

Hochschule Fresenius University of Applied Sciences Faculty of
Economics and Media International Business School Industrial
Engineering and International Management Cologne Campus
Analyses Of LNG Plant Constructions Engineering Deliverables
Progress, Planned & Forecast

Ashifuddin Mandal(MNo.400330066) & Maryam Mehdipour (MNo.400324580)

2023-07-24

Word count : 9310

email : mandal.ashifuddin@stud.hs-fresenius.de & mehdipour.maryam@stud.hs-fresenius.de

Data Science for Business

Prof.Dr. Huber

Projects work distribution to group

Ashifuddin Mandal (400330066) – Shorting the data set, Programming to get the curve of Plan, R markdown, report content are shared

Maryam Mehdipour (400324580)- Programming to get the curve of Forecast & Actual, Plotting the graph, report content are shared

Abstract

The EPC building companies that are building LNG plants have a complex organizational structure, where different departments of engineering teams are working for the completion of engineering part of the project like Civil, Mechanical, Electrical Instrumental, Safety etc. When you're expected to handle more than 1000 deliverable documents, things get extremely challenging in terms of project management. To keep the project on its original plan, it is important to keep track of any delays and see how far the project has come. This process will set up a way to figure out how far along the technical work is at each month based on the size of the task or document and the supply. To figure out how far engineering progress has reached. Progress evaluation is a structured way to check how well the project is meeting its goals and objectives. It is important for keeping the project in line with company goals and making sure the project is finished successfully. It includes comparing the expected progress with the real progress and the predicted slope. Through this method, project partners can find out what went well, what went wrong, and where there are chances for change.

Introduction

In the dynamic organizational structure of EPC (Engineering, Procurement, and Construction) construction contractors working on LNG (Liquified Natural Gas) plants, efficient project management becomes more difficult when the number of documents to be managed exceeds 1,000. In such complex situations, sustaining the project's baseline schedule becomes crucial, necessitating the capacity to monitor any delays and visualize the attained progress. This procedure seeks to establish a comprehensive method for calculating the progress of engineering work across multiple levels, based on the scope of the work and the availability of materials. Utilizing a weight decomposition concept and a technique known as progress appraisal, this method facilitates the calculation of engineering progress at both the overall and level-specific levels.

Multiple disciplines, including electrical, mechanical, civil, safety, etc., are involved in the construction of LNG facilities. With a large number of stakeholders and voluminous documentation, it is essential to develop robust mechanisms for accurately assessing project advancement. The concept of weight breakdown plays a crucial role in this context, as it enables the allocation of specific weights to various engineering tasks based on their relative significance. This enables a more nuanced evaluation of progress, as it takes into account the varying effects of various components on the overall project schedule "Progress Measurement Procedure" (n.d.). To calculate the engineering project's overall progress, the weight decomposition concept is applied to each level of work. These levels may consist of engineering design, procurement, etc. By designating appropriate weights to the various levels, the importance of each stage is reflected, allowing for a comprehensive, holistic evaluation of progress. Progress evaluation is the technique used to evaluate the project's development at each level. This methodology involves monitoring and comparing the actual progress made to the intended schedule. Deviations from the baseline schedule can be identified and quantified through meticulous record-keeping and meticulous monitoring of milestones. These insights enable project managers to assess the magnitude of delays or acceleration, laying the groundwork for effective decision-making and prompt corrective action. In addition, progress evaluation provides the means to visualize the progress made. By employing the proper tools and techniques for project management, project stakeholders can generate graphical representations of the progress, such as Gantt charts and progress curves. These visualizations not only assist in determining the current status of the project, but also facilitate the communication of progress to various stakeholders, fostering transparency and facilitating well-informed decisions. For accurate variance determination and analysis during reporting intervals, it is necessary to employ a uniform method for calculating the plan, the forecast, and the actual progress Mandal (2023).

Industrial Background

TGE Gas Engineering GmbH is a major player in the Engineering, Procurement, and building (EPC) business. It focuses on the design, engineering, procurement, and building of gas processing plants, natural gas liquefaction facilities, and other energy infrastructure projects. Their business plan is built around providing full EPC services to clients in the energy and building industries. These services cover the whole lifecycle of a project, from the first idea and feasibility studies to the thorough planning, purchasing of tools and materials, building on-site, and final commissioning. TGE Gas Engineering Company uses its technical skills and years of knowledge to give its customers high-quality projects. The company wants to be the best in its field by focusing on gas processing and liquefaction methods. This expert skill is a key part of getting contracts and completing projects well, meeting each client's specific needs and making sure the project is a success. Most of the company's customers are energy companies, utilities, and other businesses in the energy and building industries. This includes big businesses, government-owned companies, and international companies that look for, make, and sell natural gas and other energy goods. Also, TGE Gas Engineering GmbH may have worked on projects abroad, which gave them access to different markets and let them meet the wants of clients all over the world.

The global position of TGE Gas Engineering GmbH is a key part of its business plan. The company can take advantage of opportunities in different areas because it can do jobs in different regions and countries. This global method makes them more competitive and gives them the chance to work on a wide range of projects in the energy and building sectors.

When it comes to contracting methods, it's possible that TGE Gas Engineering GmbH offers a range of choices that are tailored to the needs of each project and the tastes of each client. Some of these are lump-sum turnkey contracts, in which the company takes full responsibility for the job and provides it for a fixed price. They can also sign engineering, procurement, and construction management (EPCM) contracts, which give the client more control over how the project is done while using TGE Gas Engineering GmbH's knowledge and resources.

TGE Gas Engineering GmbH may put innovation and sustainability at the top of its projects in line with industry trends and government rules. By using new technologies and practices that are good for the environment, the company shows that it is committed to providing solutions that meet global environmental goals and encourage energy efficiency.

Problems with Tracking the Progress of a Project As a major player in the EPC business, TGE Gas Engineering has to work hard to keep track of project progress and predict when things might go wrong. In the past, tracking was done by hand, with spreadsheets and separate systems being used a lot. But these methods often lack accuracy, take a lot of manual work, and might not give a full picture of how a project is going Kim, Kim, and Lee (2020).

How the S-Curve Model is Useful The only way to get around the problems with standard tracking methods and get a full picture of how a project is going is to use data-driven analysis. The S-curve model has become a useful way to see and predict how a project is going. The S-curve is based on the sigmoid curve and shows how the progress of projects builds up over time. It has typical stages of slow growth, rapid acceleration, and plateauing. The main goal of this data science project for the organization is to look at the general progress, forecast, and real S-curve of deliverable engineering documentation at TGE Gas Engineering GmbH and show them in a graph. By using data-driven methods, the project hopes to improve the way projects are managed, making it easier to predict progress and giving useful information about how the project is doing Nevalainen (2015).

Data Science

Data science is a field that combines subject-matter expertise with advanced programming, sophisticated analytics, artificial intelligence (AI), machine learning, and mathematics and statistics to derive insights from an organization's data. Strategic planning and decision-making can benefit from these insights Provost and Fawcett (2013). Data science is one of the professions that is rising the fastest in every industry because to the growing number of data sources and the data they produce. As a result, it is not unexpected that the Harvard Business Review named the position of “sexiest job of the 21st century” for data scientists (link is external to IBM). Organizations depend on them more frequently to comprehend data and provide recommendations to improve company outcomes. Analysts can gain important insights thanks to the data science lifecycle, which includes a number of roles, tools, and processes Provost and Fawcett (2013).

The lifespan begins with the gathering of raw, unprocessed, organized, and unstructured data from all pertinent sources using a variety of techniques. These methods could include real-time data streaming from systems and devices, manual input, and internet data scraping. Both structured and unstructured data may be found in data sources, including social media, the Internet of Things (IoT), log files, video, audio, and consumer data. Data processing and storage: Based on the type of data to be collected, organizations must take into account various storage options because data may take on many shapes and structures. By assisting in the creation of standards for data storage and organization, data management teams support processes for analytics, machine learning, and deep learning models. The ETL (extract, transform, and load) process or other data integration technologies are used to clean, deduplicate, transform, and integrate the data during this phase. This data preparation is essential for improving the quality of the data before it is imported into a data warehouse, data lake, or other repository Larson and Chang (2016). Exploratory data analysis is used by data scientists to find biases, trends, ranges, and distributions of data values. A/B testing hypotheses are produced by this data analytics inquiry. The appropriateness of the data for use in predictive analytics, machine learning, and/or deep learning modeling projects may also be evaluated by analysts using this information. Depending on a model's accuracy, firms may start to depend on these insights when making decisions, which would enable them to scale more easily. Communicate: Insights are finally provided as reports and other data visualizations to help business analysts and other decision-makers better understand the insights and how they will affect the firm. In addition to using specific visualization tools, data scientists may also use components of computer languages for data science, such as R or Python. In the multidisciplinary field of data science, ideas and knowledge are derived from both controlled and uncontrolled data using scientific techniques, tools, processes, and systems. It employs a wide range of techniques, including big data analytics, data mining, statistical analysis, and machine learning. Data science is crucial for modern businesses because it enables them to make decisions based on data, acquire a competitive advantage, and uncover patterns and trends in their data that they weren't previously aware of. In the context of the data analysis projects for EPC contracting.

Monitoring and predicting project progress Data science makes it possible to use past project data from the EPC contracting company's files to look at and understand how projects usually move forward. Data scientists can make reliable projections based on past data by using methods like machine learning and statistical modeling. This lets project managers and other partners track work in real time and compare it to what was expected. It helps find possible delays, use resources more efficiently, and keep projects on track by taking proactive steps.

Data visualization to help make decisions Data visualization is an important part of data science because it makes it possible to show complicated information in a way that is both visually appealing and easy to understand. Interactive data visualizations can be made as part of studying S-curve data for usable documents. These graphics give people at TGE Gas Engineering GmbH an easy-to-understand look at the general progress, forecasts, and real performance. The visual details make it easier for project managers to make good decisions and find problems or areas that need attention quickly.

Improving Project Management Data science gives EPC contracting companies like TGE Gas Engineering GmbH the ability to use data-driven project management methods. By using data science in their work, they can make the best use of their resources, streamline their processes, and improve the results of their projects as a whole. With accurate forecasts of progress and real-time tracking, the company can take

steps to reduce risks and make sure projects are finished on time and on budget.

Increasing efficiency and reducing costs EPC companies can find errors and ways to save money in their projects by using data science methods. By looking at past data and making accurate predictions about how things will go, the company can improve processes and make better use of its resources. This helps the business cut costs and work more efficiently, which increases its profits and makes it more competitive in its field.

Motivation

In the EPC contracting industry, we understand the critical importance of delivering projects on time and within budget. In addition to ensuring client satisfaction, timely completion enhances our company's reputation and competitiveness. In pursuance of continuous development, we recognize the need for more effective monitoring and forecasting methods for project progress. In order to resolve this issue, we propose a data science initiative that analyzes and visualizes the overall progress, forecast, and actual S-curve of deliverable documents for our EPC projects. The EPC industry faces obstacles when attempting to accurately predict project development and compare it to actual performance. Traditional manual monitoring methods may be imprecise, delaying the identification of deviations from the expected direction. Thus, we may be unable to implement proactive measures to keep projects on track and guarantee timely delivery. In order to overcome these constraints, we intend to employ data-driven methodologies and S-curve analysis to obtain a deeper understanding of our project's development. This data science project's successful completion will have a significant influence on our project management procedures. By accurately predicting project development and comparing it to actual performance using the S-curve model, we can more proactively identify potential delays and risks. This will allow us to allocate resources more efficiently and make well-informed decisions in order to meet project deadlines. Our enhanced project management capabilities will ultimately result in cost savings, increased client satisfaction, and a strengthened market position. In the past, we relied on manual methods of conventional monitoring, but these methods have certain limitations. Manual monitoring can be time-consuming and prone to error, delaying the identification of performance deviations. With a data-driven approach and S-curve analysis, we can tackle these limitations and develop the potential of our historical data to improve monitoring and forecasting of project progress. This data science project represents a crucial opportunity for the organization of TGE to transform our project management procedures. By leveraging the power of data-driven methodologies and S-curve analysis, we hope to improve project monitoring, optimize resource allocation, and ultimately deliver projects more efficiently and effectively. This project corresponds with our dedication to continuous development and innovation, and we are enthusiastic about its potential positive impact on our company's success in the dynamic EPC contracting industry.

Analysis goal

This project's framework is made up of four separate parts that are meant to help handle and track the progress of an engineering project. The first step is to figure out how much of a monthly deliverables is needed. Taking into account the general scope of the work and the available resources, project managers must set goals for each month that are practical and doable. These goals take into account things like the difficulty of the job, the availability of resources, and project dates. By setting these monthly goals for success, the team has a clear plan for how to move forward and can stay focused on their goals. Which is recorded in the database and we can extract that data set for the reporting works.

After the monthly goals for progress are set, the next step in the framework is to make a full plan calendar. Based on when the plan starts and when it ends, this calendar shows all the important goals, tasks, and activities. The plan schedule gives the project team a thorough plan of what they need to do and in what order. This helps them understand how jobs depend on each other. It also gives the team a way to keep track of their work and make sure they stay on track. With a well-organized plan calendar, the team can arrange their work and keep the project going along smoothly.

The third part of the strategy is keeping track of how the project is really going. As the project moves forward, it is important to keep track of and write down what has been done each month. This means keeping track of the tasks that have been finished, the goals that have been reached, and any changes from the original plan. By comparing the real progress to the plan, project managers can make smart choices and fix things if the project isn't going as expected. This information is also helpful for figuring out where the project could be improved and figuring out how well it is being done.

The fourth part of the strategy is to predict the progress curve based on the real progress data. By looking at the real work and any changes that have been made to the plan, a prediction curve is made to guess how the project will change over the next few months. This predicted curve gives project partners important information about possible problems and opportunities, which helps them make smart choices about how to use resources and make any necessary changes to the plan.

When data from the real progress and the estimated chart are put together, it's possible to do a full study of the project's general progress and any possible delays. When project managers compare what has been done to what was expected, they can see where the project is ahead or behind schedule. This study helps you make smart choices about where to put resources, how to change the plan, and how to deal with possible risks.

There are special ways to figure out how far along a project is in engineering generally and at each level or stage. One of these methods is the "weight breakdown," which includes giving different parts of the job different amounts of value, or "weight." This ranking makes it easier to figure out how important success is in each area and gives a more complete picture of how the whole project is going.

Progress evaluation is a structured way to check how well the project is meeting its goals and objectives. It is important for keeping the project in line with company goals and making sure the project is finished successfully. It includes comparing the expected progress with the real progress and the predicted slope. Through this method, project partners can find out what went well, what went wrong, and where there are chances for change. This evaluation method is a key part of helping project teams and decision-makers change strategies, make good use of resources, and make sure that projects turn out well. In the end, this comprehensive framework, which includes monthly progress goals, plan calendars, actual progress tracking, forecast curves, weight breakdowns, and methods for evaluating progress, gives project teams the tools they need to manage and analyze engineering projects well, which leads to successful and on-time project delivery Mandal (2023).

Terms and Abbreviations

The definition of common terms is set forth in Contract between Client and TGE Gas Engineering GmbH for the scope of work defined in Attachments, Scope of Work 15613/TH21/TCN/0000/0001 as set for the CONTRACT Detail Engineering of Two Full Containment LNG Storage Tanks and Pre-Commissioning / Commissioning Assistance for the LNG Coastal Deposit to be constructed and located in the industrial area of Ravenna, Italy “Progress Measurement Procedure” (n.d.). Hereinafter the more used definitions are reported

Subcontractor means TGE Gas Engineering GmbH whom has been awarded by the Contractor for a certain portion of work for the Contract

Contract means the provision set forth in the signed agreement between CONTRACTOR and SUBCONTRACTOR and all its annexes and all subsequent amendments agreed to in written by parties which shall together constitute the Contract and the term “CONTRACT” construed accordingly

Work means Detail Engineering of the LNG Tanks and Pre-Commissioning / Commissioning Assistance. The scope of work as set out in Article 2 – Attachments of the CONTRACT.

Project means design, manufacture, assembly, erection, inspection, testing, pre-commissioning, commissioning, drying, purging, cool-down and the relevant documentation of n.2 Full Containment LNG Storage Tanks, for the LNG Coastal Deposit to be constructed and located in the industrial area of Ravenna, Italy.

Weight Factor means the value of each WBS component in percent.

Progress means a figure indicates WORK improvement in percent.

WBS means Work Breakdown Structure which comprise as a hierarchy structure which describes WORK scope to identifiable packages level by level.

IFR Issued For Review, when the deliverable document is send for initial approval.

IFU Issued For Use, When the deliverable document is send for final approval.

IFC Issued For Construction, When the deliverable document is send to use for construction.

Plan- A plan will be done at the beginning of the project with the planned dates of issuing the documents. This plan will be used to build up the base line for project planning. This plan remains the same across the project, unless TGE is allowed to be modify it under the contract or an agreement with the client is achieved to do so.

Actual- The actual progress will be calculated monthly based on the status of the engineering documentation by end of the month. The achieved progress will be benchmarked with the planed progress for that month, any deviations are to be analysed and countermeasures undertaken whenever reasonable and possible.

Forecast- The forecast defines the expected completion of the engineering from the actual progress figure to the project completion. If the project is progressing as planned the forecast will be the same as the plan. In case of any deviations from the planned progress, the forecast progress will represent the expected completion for the project that will define the expected progress in the consecutive months until the project is completed.

Structure

This project is made up of four important parts that all work together to make it easy to handle and analyze an engineering project. The first step is to figure out how much growth needs to be made each month. Project managers and partners work together to set goals for success that can be reached and can be measured. These goals are important to make sure the project stays on track and moves forward so it can be finished on time. Also, having regular goals that can be measured makes it easier to check on the health of the project and spot any problems early on.

Date-related factors are used in the project, such as Plan, Forecast, and Actual. The Plan is the basic plan and timeline that was set at the start of the project. It shows how activities and goals will be done in order. On the other hand, the Forecast is a moving estimate of how the project will go based on how far it has come so far. By keeping the Forecast up-to-date all the time, the project team can make good choices and change their plans as things change. The Actual dates show the actual progress made as jobs and goals are finished during the project's completion. Part two of the task is to make a curve of plan dates based on the months. This curve shows visually how the planned tasks and goals of the project are spread out over time. It helps the project team and partners see the plan, figure out when the most important times are, and control the pace of the project to make sure it goes smoothly. The third part is about making a graph of the times of the Forecast based on the months. This curve was made by looking at how far along the project was up to a certain point and then predicting how far along it would be in the next few months. The Forecast curve gives important information about possible problems, delays, or speedups in the project's timeline. This lets you make decisions and allocate resources in a responsible way. The fourth and final part of this task is to make a model of actual progress. During the project's lifecycle, this chart shows in a visual way what has been done and how far the project has come. When project managers compare the Actual success curve to the Plan and Forecast curves, they can see how well the project is doing compared to its original plan and the direction they expected it to take. Any differences between the shapes can show areas that need to be looked at or optimized to prevent delays and improve the project's general performance. By putting together all the information from the Plan, Forecast, and Actual progress charts, the project team can do a full study of the project's progress and delays. This study not only lets you know where the project is right now, but it also helps you find possible risks and bottlenecks. With this knowledge, project managers can make smart choices, change deadlines or the way resources are used if they need to, and guide the project toward its successful end. The structure of this project facilitates successful project management to be carried out via the use of monthly progress objectives, dynamic forecasting, and real-time monitoring of progress. The capacity of the project team to see patterns, arrive at choices based on data, and proactively adapt to obstacles and changes throughout the lifespan of the project is improved when the team makes use of visual curves to represent Plan, Forecast, and Actual progress. In the end, this strategy makes sure that engineering projects are completed on time and within the scope that has been set, therefore satisfying the expectations of all stakeholders who are involved Mandal (2023).

Methodology

In this project, the calculation of progress is based on a set of document deliverables that contain all contractually mandated engineering project deliverables. Each deliverable is designated a weighting factor spanning from 8 to 400 (can be found in dataset as variable Manhours), which reflects the document's level of complexity. The greater the weighting factor, the more complex and difficult the document is to create. This weighting system is implemented to accurately reflect the effort required for each deliverable and provide an accurate assessment of the overall complexity of the project Mandal (2023). To determine the overall weighted progress of the project, all document deliverables' weighting points are added together. This total represents the total scope of the engineering work. The aggregate weighted progress provides a comprehensive measure of the project's advancement by considering the complexity of each deliverable through its designated weighting factor@mandal_engineering_2023. Each deliverable document is associated with two significant dates: the date it is issued for review (IFR) and the date it is issued for use (IFU). On the basis of these two milestones, the realized development for each deliverable is computed. 70% of a document's total weight is deemed to be accomplished when it is first released for review. This represents a significant step toward completion, as it indicates progress and the readiness to receive input and feedback from relevant client engineers Mandal (2023).

When a document is issued for use (IFU), the final phase of development for a deliverable is completed. At this point, the document is considered complete and entirely conforms to its intended purpose's specifications. As a result, when the document obtains IFU status, it carries 100 percent of its total weight.

The cumulative progress of all document deliverables contributes to the engineering project's overall advancement. This method provides a nuanced and objective assessment of the project's progress by taking into consideration both the complexity of each deliverable and its corresponding milestones.

This method of calculating progress allows project managers to monitor the project's performance, identify potential delays or obstacles in document preparation, and effectively allocate resources. In addition, it provides a clear picture of the engineering project's development at any given time, facilitating decision-making and ensuring contractual obligations are met in a timely and effective manner Mandal (2023).

Key Milestone	Progress %	Comment
1st Issue Issued for Review (IFR)	70%	TGE shall submit first edition of document with proper technical content for proceeding the work, and the initial progress (70%) is recognised in this first issue.
Issue for Use (IFU)	100%	Once document is reviewed (whenever applicable) and the content is adequate to be sent as final, it will be Issued for Use and the additional 30% progress will be recognised.

Figure 1: Progress Methodology

Data Set

It is hard to overstate the significance of a data set in a data science project, as it functions as the basis for the entire endeavor. The quality and precision of the derived results and insights are directly dependent on the carefully constructed data set. The analysis performed on the data set impacts decision-making and problem-solving. In addition, the accuracy of machine learning algorithms and statistical models is highly dependent on the data set's quality and representativeness. Consequently, a well-prepared data set results in more robust and accurate models, facilitating data-driven decisions that have a significant impact on the project's objectives Grolemond (n.d.).

To ensure the integrity of the results, it is essential to generate the data set with care and without errors. The “Garbage In, Garbage Out” (GIGO) principle emphasizes the direct relationship between data quality and the outcome of an analysis. Any analysis or model based on defective data will produce erroneous interpretations and skewed conclusions. Misinformation within the data set can have severe repercussions, particularly when crucial decisions are dependent on the analysis@grolemond_welcome_nodate.

In addition, data cleansing is a crucial step that plays a pivotal role in data science initiatives. During the data preprocessing phase, data cleansing entails identifying and rectifying errors, inconsistencies, and inaccuracies in the data set. This process includes duties such as dealing with missing values, removing duplicates, standardizing formats, detecting and dealing with outliers, and normalizing and scaling numerical characteristics. The integrity of the analysis, the precision of the results, and the overall quality of the data science endeavor are enhanced by data cleansing.

Data scientists and researchers can considerably improve the results of a project by devoting time and effort to meticulously creating and cleansing the data set. An accurate and informative data set enables more robust analysis, improved models, and data-driven decisions that inspire stakeholders' confidence and trust. In addition, a well-documented and trustworthy data set promotes transparency and reproducibility, allowing other researchers to validate the findings and ensuring that the data science endeavor contributes positively to the advancement of knowledge and practical applications. The meticulous management of data collections is a fundamental pillar of successful and influential data science initiatives.

In the TGE gas engineering organization, Pims R4, which is based on the Omega 360 document management system, is used to produce data sets. This software automates the monitoring of variables such as Actual dates, resulting in more precise results. In addition, there are variables such as Plan dates that require manual input at the start of the undertaking. The combination of automated monitoring and manual data entry guarantees comprehensive data collection and administration throughout the lifecycle of the project.

The Omega 360 document management software has a lot of features for keeping track of documents efficiently throughout the span of a project. The document record in the program acts as a central location for all project-related papers, including those from providers. It keeps multiple versions of documents, making sure that end users always have access to the latest version and letting them get to older versions when they need to. The software lets you look based on information and file content, which makes it easy to find documents that need to be taken care of. Contractors gain from an easy-to-use interface that lets them send documents and view them through web pages. The review and acceptance of documents are made easier with a sharing grid, email alerts, and marking tools. This makes it easier for people to work together. Omega 360 also makes it easy to create uniform documents by using MS Office templates, auto-numbering documents, and distributing documents based on customizable grids. The software also keeps track of communication and lets you make decisions based on a variety of adjustable reports. Omega 360 improves document handling, teamwork, and planning with its easy-to-use layout and powerful features, which eventually helps a project succeed.

```
##{r} library(knitr) library(kableExtra) library(readxl) library(dplyr) dataSet <-  
read_excel("C:/Users/MandalA/Downloads/project/dataSet.xlsx") kable(dataSet) ##
```

Data Preparation & Coding

Setup and Data Import: When opening R Studio first the libraries lubridate, hms, ggplot2 and tidyverse and its sub-libraries, which contain important functions for processing, analyzing and visualizing data, must be loaded. This line, `library(readxl)`, gets the “readxl” package. With the “readxl” package, data from Excel files (.xlsx and .xls) can be read into R data frames. It has methods like `read_excel()` that let you input information from Excel files.

This line, `library(dplyr)`, gets the “dplyr” package. The “dplyr” package is a famous and powerful way to work with statistics in R. It gives you a set of tools that make it easy to filter, sort, analyze, and change data. “dplyr” has many useful methods, such as `filter()`, `order()`, `edit()`, `summarize()`, `group_by()`.

```
# Importing the required packages
library(readxl)
library(dplyr)
```

```
##
## Attache Paket: 'dplyr'

## Die folgenden Objekte sind maskiert von 'package:stats':
##
##      filter, lag

## Die folgenden Objekte sind maskiert von 'package:base':
##
##      intersect, setdiff, setequal, union
```

With the `data.frame()` function, R creates a new data frame. A data frame is a tabular data structure with two dimensions that can hold different kinds of data. It's like a spreadsheet or a table.

`date_month = "2022-01", "2022-02", ..., "2024-04"`: Here, we are adding a new field to the data frame called `date_month`. The `c()` function, which is used to put together a vector, is used to give the numbers for this column. In this case, the vector is made up of a series of “YYYY-MM” date lines, each of which stands for a month.

```
dummydf <- data.frame (date_month = c("2022-01",
                                       "2022-02",
                                       "2022-03",
                                       "2022-04",
                                       "2022-05",
                                       "2022-06",
                                       "2022-07",
                                       "2022-08",
                                       "2022-09",
                                       "2022-10",
                                       "2022-11",
                                       "2022-12",
                                       "2023-01",
                                       "2023-02",
                                       "2023-03",
                                       "2023-04",
                                       "2023-05",
                                       "2023-06",
```

```

"2023-07",
"2023-08",
"2023-09",
"2023-10",
"2023-11",
"2023-12",
"2024-01",
"2024-02",
"2024-03",
"2024-04"
))

```

This function comes from the “readxl” package, which is used to read data from Excel files (.xlsx and.xls) into R data frames. Its input is the path to the Excel file, and it gives a data frame with the data from the Excel file. In this case, “C:/Users/MandalA/Downloads/project/dataSet.xlsx” is where you can find the Excel file. `dataSet`: This is the name of the data frame where the information from the Excel file will be stored. This data frame will be given the data from the Excel file.

```

dataSet <- read_excel("C:/Users/MandalA/Downloads/project/dataSet.xlsx")
print(dataSet)

```

```

## # A tibble: 564 x 21
##   'Document ID'      Title      'Current Revision' Current Client Revis-1
##   <chr>             <chr>      <chr>             <chr>
## 1 C5663/CS30/CAL/4500/3101 Static Ca~ <NA>             <NA>
## 2 C5663/CS30/CAL/4500/3201 Static Ca~ B             E02
## 3 C5663/CS30/CAL/4500/3301 Static Ca~ <NA>             <NA>
## 4 C5663/CS30/CAL/4500/3401 Static Ca~ 1             EA1
## 5 C5663/CS30/CAL/4500/3501 Static Ca~ 2             EF1
## 6 C5663/CS30/CAL/4500/3601 Static ca~ 2             EF1
## 7 C5663/CS30/CAL/4500/3701 Static Ca~ <NA>             <NA>
## 8 C5663/CS30/CAL/4500/3702 Static Ca~ <NA>             <NA>
## 9 C5663/CS30/DWG/4500/3101 Steel Str~ <NA>             <NA>
## 10 C5663/CS30/DWG/4500/3102 Steel Str~ <NA>             <NA>
## # i 554 more rows
## # i abbreviated name: 1: 'Current Client Revision'
## # i 17 more variables: 'Client Document ID' <chr>, 'Document Purpose' <chr>,
## #   'Client Action' <chr>, 'Client Review Status' <chr>, Discipline <chr>,
## #   'Man Hours' <dbl>, Responsible <chr>, Keywords <lgl>, 'Client WBS' <chr>,
## #   'Document Need Translation' <lgl>, 'Work Pack ID' <lgl>,
## #   'IFR Planned' <dtm>, 'IFR Forecast' <dtm>, 'IFR Actual' <dtm>, ...

```

This line makes a new column in the `dataSet` data frame called “weight” to hold the calculated weights.

`dataSet$Man Hours`: This part talks about the “Man Hours” field in the ‘`dataSet`’ data frame. Backticks () are put around the column name because it has a space in it. Using backticks helps R understand what the column name means.

`sum(dataSet$ManHours)`: This adds up all the numbers in the `dataSet` data frame’s “ManHours” field. (`dataSet$ManHours / sum(dataSet$ManHours)`): This works out how much each “Man Hours” number contributes to the whole. It basically figures out how much each “Man Hours” amount adds to the total number of “Man Hours.”

- 100: Multiplying the proportion by 100 turns it into a percentage, which gives the weight of each “Man Hours” number as a percentage of the overall “Man Hours.”

```

dataSet$weight <- round((dataSet$`Man Hours`/sum(dataSet$`Man Hours`))*100, digits = 4)

dataSet$weightIFR <- round((dataSet$`Man Hours`/sum(dataSet$`Man Hours`))*70, digits = 4)

dataSet$weightIFU <- round((dataSet$`Man Hours`/sum(dataSet$`Man Hours`))*30, digits = 4)

```

These lines of code are written in R and use the “Man Hours” column in the `dataSet` to figure out the weights for each field.

`%>%`: In R, the pipe operator (`%>%`) is used to link together several processes. It uses the result of the last operation as the first input for the next operation. This line tells the data in the `dataSet` data frame to be grouped by the “Discipline” field. This means that the next actions will be done on each group of data separately, based on the numbers in the “Discipline” column.

`change(WeightPerDiscipline = sum(Man Hours))`: This line adds a new field called “WeightPerDiscipline” to the `dataSet` data frame. For each row, it figures out the total number of “Man Hours” in each group (discipline) and puts that number in the “WeightPerDiscipline” column for that row. It basically adds up the number of “Man Hours” for each field. This is a reference to the “WeightPerDiscipline” field in the `dataSet` data frame, which was made in the last step.

`(dataSet$ManHours/dataSet$WeightPerDiscipline)`: This works out how many “Man Hours” are in each row compared to how many “Man Hours” are in the discipline as a whole. It basically figures out how much each “Man Hours” worth in its field adds to the whole.

```

dataSet <- dataSet %>%
  group_by(Discipline) %>%
  mutate(WeightPerDiscipline = sum(`Man Hours`))

dataSet$WeightPerDiscipline <- round((dataSet$`Man Hours`/ dataSet$WeightPerDiscipline)*100, digits = 3)

```

Aggregating IFR based on month: The code uses the `aggregate()` function to add up the “weightIFR” numbers in the “IFR Planned” column of the `dataSet` data frame based on the month. It changes the name of the first column in the data frame to “date_month.”

Aggregating IFU by month: The code uses the `aggregate()` function to add up the “weightIFU” numbers in the “IFU Planned” column of the `dataSet` data frame based on the month. It changes the name of the first column in the data frame to “date_month.”

Merging IFR and IFU data frames: The code combines the combined IFR and IFU data frames based on the “date_month” column by using the `merge()` function with the `all = TRUE` option. This makes sure that all months are included in the result.

Using `is.na()` and assignment, it fills in any NA (missing) numbers with 0. It adds up the “weightIFR” and “weightIFU” numbers to get the “totalPlanned” column. Merging with a dummy data frame and more data manipulation: The code combines the data frame from the previous step with a fake data frame (`dummydf`) based on the “date_month” column using the `merge()` function with the `all = TRUE` option. This makes sure that all months are included in the result.

Using `is.na()` and assignment, it fills in any NA (missing) numbers with 0. The `select()` function is used to choose only the “date_month” and “totalPlanned” fields. Using the `modify()` method and `cumsum()`, it figures out the “totalRunningPlan” column, which is the sum of “totalPlanned” over time. With the `select()` method, it only picks the “date_month” and “totalRunningPlan” fields.

```

#Aggregating IFR based on month
dfp_IFR <- aggregate(weightIFR ~ format(as.Date(dataSet$`IFR Planned`), "%Y-%m"), dataSet, sum)
colnames(dfp_IFR)[1] = "date_month"

```

```

#Aggregating IFU
dfp_IFU <- aggregate(weightIFU ~ format(as.Date(dataSet$`IFU Planned`), "%Y-%m"), dataSet, sum)
colnames(dfp_IFU)[1] = "date_month"

dfp_IFR_IFU <- merge(x=dfp_IFR,y=dfp_IFU, by = c("date_month"),all = TRUE)
dfp_IFR_IFU[is.na(dfp_IFR_IFU)] <- 0

dfp_IFR_IFU$totalPlanned <- dfp_IFR_IFU$weightIFR+dfp_IFR_IFU$weightIFU

dfp_IFR_IFU_planned <- merge(x=dummydf,y=dfp_IFR_IFU, by = c("date_month"),all = TRUE)

dfp_IFR_IFU_planned[is.na(dfp_IFR_IFU_planned)] <- 0

dfp_IFR_IFU_planned <- select(dfp_IFR_IFU_planned ,date_month, totalPlanned)

dfp_IFR_IFU_planned <- dfp_IFR_IFU_planned %>%
  mutate(totalRunningPlan = cumsum(totalPlanned))

dfp_IFR_IFU_planned <- select(dfp_IFR_IFU_planned ,date_month, totalRunningPlan)

```

Adding up the IFR based on the month: The code uses the aggregate() function to group the “IFR Forecast” column data by month and find the sum of the “weightIFR” numbers for each group. The result is saved in the dff_IFR file, and the first field is now called “date_month.”

Adding up the IFU by month: Similar to step 1, the code uses the aggregate() function to group the data by month from the “IFU Forecast” column in the dataSet data frame and figure out the sum of the “weightIFU” values. The result is saved in dff_IFU, and the first column’s name is changed to “date_month.”

When IFR and IFU data frames are combined: The code combines the dff_IFR and dff_IFU data frames based on the “date_month” field by using the merge() function with all = TRUE to include all months in the result.

Using is.na() and assignment, it fills in any NA (missing) numbers with 0. It adds the “weightIFR” and “weightIFU” fields to figure out the “totalForecast” column. Merging

with a fake data frame and doing more with the data: The code combines the result of the previous step’s data frame with the dummydf data frame based on the “date_month” column. This is done with the merge() function and the all = TRUE parameter to include all months in the result. Using is.na() and assignment, it fills in any NA (missing) numbers with 0. With the select() method, it only picks the “date_month” and “totalForecast” values.

It uses the modify() method and cumsum() to figure out the “totalRunningForecast” column, which is the sum of “totalForecast” over time. With the select() method, only the “date_month” and “totalRunningForecast” variables are chosen.

```

#Aggregating IFR based on month
dff_IFR <- aggregate(weightIFR ~ format(as.Date(dataSet$`IFR Forecast`), "%Y-%m"), dataSet, sum)
colnames(dff_IFR)[1] = "date_month"

#Aggregating IFU
dff_IFU <- aggregate(weightIFU ~ format(as.Date(dataSet$`IFU Forecast`), "%Y-%m"), dataSet, sum)
colnames(dff_IFU)[1] = "date_month"

dff_IFR_IFU <- merge(x=dff_IFR,y=dff_IFU, by = c("date_month"),all = TRUE)
dff_IFR_IFU[is.na(dff_IFR_IFU)] <- 0

```

```

dff_IFR_IFU$totalForecast <- dff_IFR_IFU$weightIFR+dff_IFR_IFU$weightIFU

dff_IFR_IFU_Forecast <- merge(x=dummydf,y=dff_IFR_IFU, by = c("date_month"),all = TRUE)

dff_IFR_IFU_Forecast[is.na(dff_IFR_IFU_Forecast)] <- 0

dff_IFR_IFU_Forecast <- select(dff_IFR_IFU_Forecast ,date_month, totalForecast)

dff_IFR_IFU_Forecast <- dff_IFR_IFU_Forecast %>%
  mutate(totalRunningForecast = cumsum(totalForecast))

dff_IFR_IFU_Forecast <- select(dff_IFR_IFU_Forecast ,date_month, totalRunningForecast)

```

Adding up the IFR based on the month: The code uses the aggregate() function to group the “IFR Actual” column data by month and find the sum of the “weightIFR” numbers for each group. The result is saved in the dfa_IFR file, and the first field is now called “date_month.”

Adding up the IFU by month: Similar to step 1, the code uses the aggregate() function to group the data by month from the “IFU Actual” column in the dataSet data frame and figure out the sum of the “weightIFU” values. The result is saved in the dfa_IFU file, and the first field is now called “date_month.”

When IFR and IFU data frames are combined: The code combines the dfa_IFR and dfa_IFU data frames based on the “date_month” field by using the merge() function with all = TRUE to include all months in the result. Using is.na() and assignment, it fills in any NA (missing) numbers with 0. It adds the “weightIFR” and “weightIFU” fields to figure out the “totalActual” column.

Merging with a fake data frame and doing more with the data: The code combines the result of the previous step’s data frame with the dummydf data frame based on the “date_month” column. This is done with the merge() function and the all = TRUE parameter to include all months in the result. Using is.na() and assignment, it fills in any NA (missing) numbers with 0.

With the select() method, it only picks the “date_month” and “totalActual” values. It uses the modify() method and cumsum() to figure out the “totalRunningActual” column, which is the sum of “totalActual” over time. With the select() method, it only picks the “date_month” and “totalRunningActual” values.

```

#Aggregating IFR based on month
dfa_IFR <- aggregate(weightIFR ~ format(as.Date(dataSet$`IFR Actual`), "%Y-%m"), dataSet, sum)
colnames(dfa_IFR)[1] = "date_month"

```

```

#Aggregating IFU
dfa_IFU <- aggregate(weightIFU ~ format(as.Date(dataSet$`IFU Actual`), "%Y-%m"), dataSet, sum)
colnames(dfa_IFU)[1] = "date_month"

dfa_IFR_IFU <- merge(x=dfa_IFR,y=dfa_IFU, by = c("date_month"),all = TRUE)
dfa_IFR_IFU[is.na(dfa_IFR_IFU)] <- 0

dfa_IFR_IFU$totalActual <- dfa_IFR_IFU$weightIFR+dfa_IFR_IFU$weightIFU

dfa_IFR_IFU_Actual <- merge(x=dummydf,y=dfa_IFR_IFU, by = c("date_month"),all = TRUE)

dfa_IFR_IFU_Actual[is.na(dfa_IFR_IFU_Actual)] <- 0

dfa_IFR_IFU_Actual <- select(dfa_IFR_IFU_Actual ,date_month, totalActual)

dfa_IFR_IFU_Actual <- dfa_IFR_IFU_Actual %>%

```



```
mutate(totalRunningActual = cumsum(totalActual))

dfa_IFR_IFU_Actual <- select(dfa_IFR_IFU_Actual ,date_month, totalRunningActual)
```

In the first part of the code, the `dfp_IFR_IFU_planned` and `dff_IFR_IFU_Forecast` data sets are brought together. To combine the first two data frames, the `merge()` method is called twice with `df_final` as the first option. The “date_month” column is used to do the merge, and the “all = TRUE” argument makes sure that all months from both data frames are included in the result, even if some months have missing values (NA). The `df_final` data frame holds the result of the merge.

Next, the code merges the new `df_final` data frame with the `dfa_IFR_IFU_Actual` data frame. To join the third data frame, the `merge()` method is called a third time with `df_final` as the first argument. Again, the merger is based on the “date_month” column, and the “all = TRUE” argument makes sure that all months from all three data frames are included in the end result, even if some months in any of the data frames have missing values (NA). The result of the merge is put back into the `df_final` data frame, which now has all of the data from the three data frames in one place.

After the code is run, the data from the `dfp_IFR_IFU_planned`, `dff_IFR_IFU_Forecast`, and `dfa_IFR_IFU_Actual` data frames will be put into the `df_final` data frame. It has fields for “date_month,” “totalRunningPLan” (cumulative total of planned weights), “totalRunningForecast” (cumulative total of predicted weights), and “totalRunningActual” (cumulative total of actual weights) for each month. The data has now been combined and is ready to be analyzed or shown visually. This gives a full picture of the planned, predicted, and real weights over time.

```
df_final <- merge(x=dfp_IFR_IFU_planned,y=dff_IFR_IFU_Forecast, by = c("date_month"),all = TRUE)
df_final <- merge(x=df_final,y=dfa_IFR_IFU_Actual, by = c("date_month"),all = TRUE)
```

Visualisation

Using data from the `df_final` data frame, the R code uses the `ggplot2` library to make a line plot with points that shows the sum of planned, predicted, and real weights over time. When the `ggplot` method is run for the first time, `df_final` is used as the data source. On the plot, the combined totals will be shown on the y-axis, and the “date_month” numbers will be shown on the x-axis. Two geometry shapes, `geom_line()` and `geom_point()`, are added to the plot to show the “Plan” data. The x-axis (`date_month`) and y-axis (`totalRunningPlan`) values for the “Plan” data are mapped by the `aes()` function inside the mapping function. The `group = 1` argument makes sure that all of the “Plan” data points and lines are linked together in a single group. The “Plan” data line and points will be blue if the `color = “Plan”` argument is used.

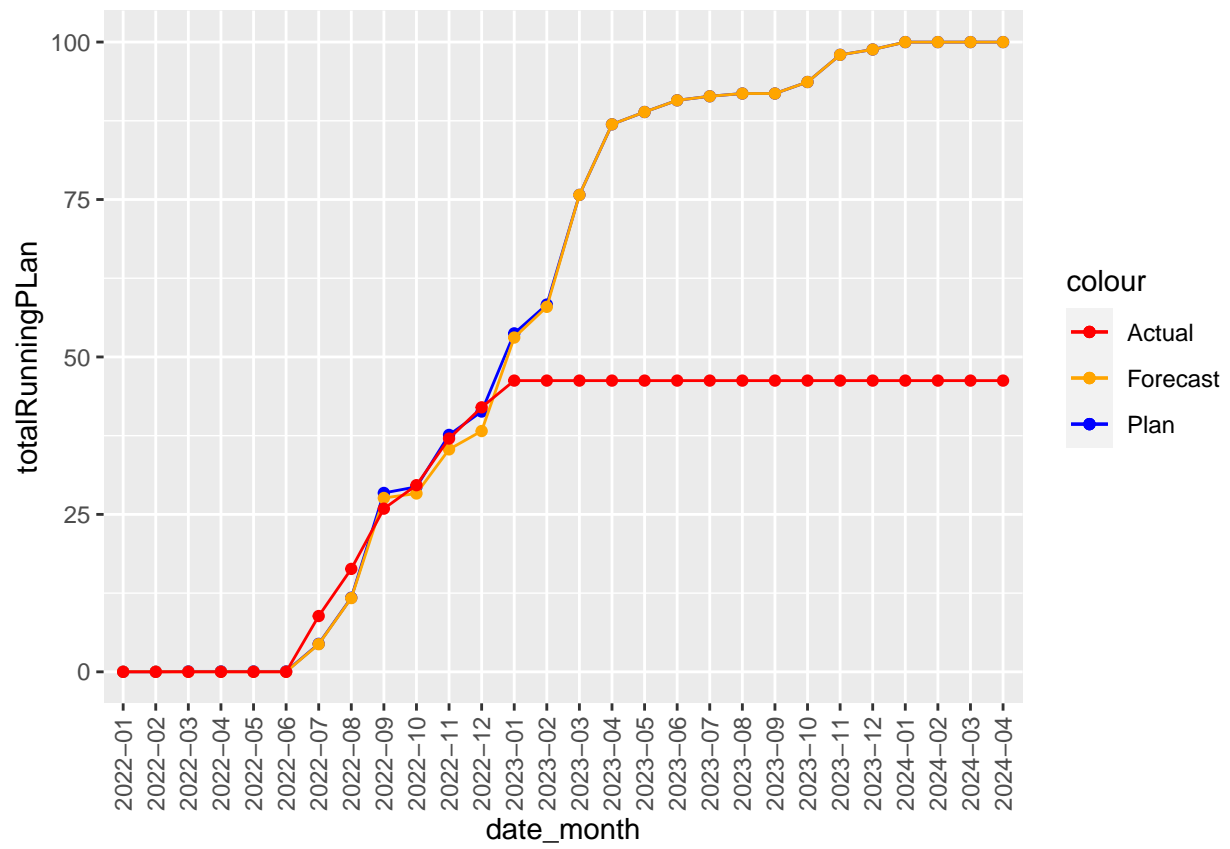
`Geom_line()` and `geom_point()` are also used to plot the “Forecast” data. The x-axis (`date_month`) and y-axis (`totalRunningForecast`) values for the “Forecast” data are mapped by the `aes()` method inside the mapping block. The `group = 1` statement puts all of the “Forecast” points and lines together. The “Forecast” data line and points will be orange if the `color = “Forecast”` argument is used. `Geom_line()` and `geom_point()` are also used to show how the “Actual” data looks. The x-axis (`date_month`) and y-axis (`totalRunningActual`) values for the “Actual” data are mapped by the `aes()` function inside the mapping function. The `group = 1` statement puts all of the “Actual” points and lines together. The “Actual” data line and points will be red if the `color = “Actual”` argument is used.

The `scale_x_discrete(guide = guide_axis(check.overlap = TRUE))` line makes the x-axis text marks appear at a 90-degree angle so that they don’t overlap. This makes the x-axis text easier to read. The `scale_color_manual()` function is used to set the colors of the lines and points by hand. Data for “Plan” is shown in blue, data for “Forecast” in orange, and data for “Actual” in red.

Lastly, the line `theme(axis.text.x = element_text(angle = 90, vjust = 0.5))` changes how the x-axis text looks by turning the names 90 degrees to make them easier to read and centering the text vertically.

```
library(ggplot2)

ggplot(data = df_final) +
  geom_line(mapping = aes(x = date_month, y = totalRunningPlan, group = 1, color = "Plan")) +
  geom_point(mapping = aes(x = date_month, y = totalRunningPlan, group = 1, color = "Plan")) +
  geom_line(mapping = aes(x = date_month, y = totalRunningForecast, group = 1, color = "Forecast")) +
  geom_point(mapping = aes(x = date_month, y = totalRunningForecast, group = 1, color = "Forecast")) +
  geom_line(mapping = aes(x = date_month, y = totalRunningActual, group = 1, color = "Actual")) +
  geom_point(mapping = aes(x = date_month, y = totalRunningActual, group = 1, color = "Actual")) +
  scale_x_discrete(guide = guide_axis(check.overlap = TRUE)) +
  scale_color_manual(values = c("Plan" = "blue", "Forecast" = "orange", "Actual" = "red")) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```



Conclusion

In the end, putting the four different parts of this project into action has shown to be a very effective way to make engineering projects easier to handle and analyze. The first part, figuring out how much progress needs to be made each month, sets clear, measurable goals that help the project team stay on track and track progress well. Using date-related factors like “Plan,” “Forecast,” and “Actual” makes it possible to keep track of all data throughout the project’s lifecycle. The project team is able to make better choices and shift strategies as things change because the Forecast curve is constantly updated. This leads to better project results. Visual images of Plan dates, Forecast dates, and Actual progress in the form of shapes show important information about the project’s schedule and major goals. These visual tools help the team find the most important times, keep track of the pace, and make the best use of resources so that the project goes smoothly. The fourth step, building a model of the project’s actual progress, lets project managers compare the project’s performance to the original plan and the predicted path. By finding areas of departure and possible delays, the team can deal with problems ahead of time and improve the general performance of the project. When the data from the Plan, Forecast, and Actual progress charts are combined, the project team gets a full picture of the project’s progress and any possible risks. This information makes it possible to make well-informed decisions, make changes to strategies at the right time, and use resources well, all of which lead to the successful end of the project. This project report’s structure shows how important it is to set clear monthly goals, use dynamic forecasting, and use visual models for tracking and analyzing progress. The team is able to finish engineering jobs on time and within the plan that was set up by the structured method. With the ability to change and react to problems in a proactive way, the project team is ready to meet the needs of all parties and make things work out well. The method described in this report can be used as a useful guide for future engineering projects. This helps to improve the field and ensure the success of large-scale projects.

Github

##Contributions using Git and GitHub I chose to contribute for the first time using Git and GitHub. To get began, I quickly and easily created a GitHub account. I installed Git on my computer by following the provided guide. Then, I located a repository to which I wished to contribute and selected the “fork” icon to create my own copy of the repository. Clicking the “code” option on my GitHub account and copying the URL allowed me to clone the forked repository to my local machine. I opened the terminal and pasted the URL before running the git clone command to create a local copy on my computer.

I navigated to the repository directory using cd and created a new branch using git switch -c your-new-branch-name. I gave the branch a descriptive name, such as “add-Stephan-Huber.” After creating the branch, I opened the “I_am_a_data_scientist.md” file in a text editor and added my name, GitHub account, and the paper I wanted to reproduce. After submitting my modifications, I ran git status to validate the changes. To include my modifications in the branch, I staged them with git add.. Then, I committed the modifications using git commit -m “Add my-name to the list” where “my-name” was replaced with my actual name.

I used the commands git config --global user.name “FIRST_NAME LAST_NAME” and git config --global user.email “MY_NAME@example.com” to set up Git Bash with my GitHub email and username. I was eager to share my contribution with the world as I felt a sense of accomplishment. I submitted my modifications to GitHub using the command git push -u origin your-new-branch-name, where “your-new-branch-name” was replaced with the name of my branch.

I finally submitted my changes to GitHub for review. I’m ecstatic to have made my first effective contribution using Git and GitHub!

Affidavit

I hereby affirm that this submitted paper was authored unaided and solely by me. Additionally, no other sources than those in the reference list were used. Parts of this paper, including tables and figures, that have been taken either verbatim or analogously from other works have in each case been properly cited with regard to their origin and authorship. This paper either in parts or in its entirety, be it in the same or similar form, has not been submitted to any other examination board and has not been published I have read the Handbook of Academic Writing by Hildebrandt & Nelke (2019) and have endeavored to comply with the guidelines and standards set forth therein. I acknowledge that the university may use plagiarism detection software to check my thesis. I agree to cooperate with any investigation of suspected plagiarism and to provide any additional information or evidence requested by the university. I assure the project report contains the following:

The report contains, 5000-6000 words, a title and personal detail (name, email, matriculation number), a word count, an abstract, a bibliography that was created using BibTeX applying an citation style, the complete R code that is necessary to replicate your results, detailed information on how the data are downloaded and read in to R, an introduction where you guide the reader and a conclusion where you summarize, your work and discuss what you would work on if you would have more time available, all significant resources that were used to write the report and the R code, the filled out Affidavit, a brief description of the successful contribution using Git and GitHub as explained,

The submission of the project contains. . . the .Rmd file of the report, the .pdf file of the report, the .html file of the report, all files necessary that are not available online to reproduce the report and the R code therein, the .Rmd file of the presentation, the .html file of the presentation,

Ashifuddin Mandal & Maryam Mehdipour, 24.07.2023, Köln

References

- Grolemund, Hadley Wickham and Garrett. n.d. *Welcome / R for Data Science*. Accessed July 24, 2023. <https://r4ds.had.co.nz/>.
- Kim, Seungho, Sangyong Kim, and Dong-Eun Lee. 2020. "Sustainable Application of Hybrid Point Cloud and BIM Method for Tracking Construction Progress." *Sustainability* 12 (10): 4106. <https://doi.org/10.3390/su12104106>.
- Larson, Deanne, and Victor Chang. 2016. "A Review and Future Direction of Agile, Business Intelligence, Analytics and Data Science." *International Journal of Information Management* 36 (5): 700–710. <https://doi.org/10.1016/j.ijinfomgt.2016.04.013>.
- Mandal, Ashifuddin. 2023. "Engineering Progress Measurement Procedure." 62000/ TH42/PCD/ 0000/0001. TGE Gas Engineering. https://docs.google.com/document/d/1SbyOq3ucPuJaMiZp-AIIAqqcPC5bHkAN/edit?usp=drive_link&ouid=100249403080617006590&rtpof=true&sd=true.
- Nevalainen, Terttu. 2015. "Descriptive Adequacy of the S-Curve Model in Diachronic Studies of Language Change." *Studies in Variation, Contacts and Change in English* 16. <https://varieng.helsinki.fi/series/volumes/16/abstracts.html#nevalainen>.
- "Progress Measurement Procedure." n.d. 62000/ TH42/PCD/ 0000/0001. TGE Gas Engineering. https://docs.google.com/document/d/1kUHw3Yk6GztNRU2CSVY8I__2zg9lY-hT/edit?usp=drive_link&ouid=100249403080617006590&rtpof=true&sd=true.
- Provost, Foster, and Tom Fawcett. 2013. *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. "O'Reilly Media, Inc."