**INTRODUCTION**

**1.1 Overview**

I joined XchangeData Technologies Pvt. Ltd. as a Full Stack Development Intern where I learned how an application functions and how front-end and back-end can be linked to give out a hassle-free user-friendly application. When I joined, I learned about .Net framework, JS, Wordpress and Php and their life cycles and real time use. After a few days, I was assigned to the ongoing website development. I was repeatedly given CRs to work on. I got a chance to understand the sprints that development teams work on to deliver their work on the date specified to the customers.

Apart from working on several CRs of the live project, I also solved Issues raised by the QA team. The Internship has helped me a lot to grow as a developer and understand how a product-based Company functions.

**1.2 Project Overview**

**1.2.1 Bends & Curves**

“Bends & Curves is civil structure related desktop application. It takes data from a drawing drawn in ETABS displays all data i.e name, width, depth of beam and column etc. Allows us to change the properties of beam and column. Set vertical and horizontal arrangements, Similar and mirror beam arrangements and draw drawing in Autocad accordingly ”

**1.2.2 OfficeMate**

OfficeMate - Simplifying Administrative Task. OfficeMate is a comprehensive Windows application designed to simplify administrative tasks within an organization. It provides employees with a user-friendly platform to streamline various processes and enables efficient communication between employees and the admin. The application offers several modules to cater to different administrative needs.

1. Leave Application:

2. Request Application:

3. Reimbursement Application:

4. Chat:

OfficeMate aims to enhance productivity and efficiency within the organization by streamlining administrative tasks and promoting seamless communication. By providing a centralized platform for leave applications, request submissions, reimbursement requests, and chat functionalities, the application ensures that employees and the admin can efficiently manage their tasks and interact with each other. This comprehensive approach fosters a collaborative work environment while reducing administrative burdens for both employees and the admin.

**1.3 Project Objectives**

**1.3.1 Bends & Curves**

* + - 1. Provide a user-friendly dashboard: The application should have a simple and intuitive dashboard that allows users to easily navigate and access the different features and functionalities.
      2. Display basic framework and story details: The application should present the basic framework of the drawing, showing the arrangement of beams and columns. It should also provide details about the different stories and their relationships, such as identifying similar stories and master stories.
      3. Beam design: The application should include a tab for beam design, where users can determine the minimum required area of steel (AST) and add steel bars of different diameters to fulfill the AST requirement. This feature helps ensure the beams have sufficient strength.
      4. Drawing functionality: The application should have a drawing tab that enables users to generate drawings based on the defined arrangements. Clicking the draw button should trigger the Autocad instance and automatically create drawings according to the set arrangements in the arrangement tab.
    1. **OfficeMate**

Streamline Administrative Tasks: The primary objective of OfficeMate is to simplify and streamline administrative tasks within the organization. By providing modules for leave applications, request submissions, and reimbursement requests, the application aims to automate and standardize these processes, reducing paperwork and manual effort.

Enhance Employee Experience: OfficeMate seeks to improve the overall employee experience by providing a user-friendly interface for managing various administrative tasks. By offering a centralized platform, employees can conveniently apply for leaves, submit requests, and track their statuses. This objective aims to empower employees, increase their satisfaction, and optimize their productivity.

Expedite Decision-Making: With OfficeMate, the admin can review and process leave applications, requests, and reimbursement requests in a timely manner. The objective is to facilitate faster decision-making by providing all necessary information in a structured format. This helps in minimizing delays and ensuring efficient allocation of resources.

Promote Effective Communication: The Chat module in OfficeMate aims to promote effective communication between employees and the admin. By enabling one-on-one conversations, employees can directly address their queries and concerns, while the admin can efficiently communicate with all employees. This objective emphasizes transparent and streamlined communication channels within the organization.

Increase Efficiency and Productivity: By automating administrative tasks and providing a centralized platform, OfficeMate aims to increase overall efficiency and productivity within the organization. This objective reduces the time and effort spent on manual administrative processes, allowing employees and the admin to focus on more value-added activities.

**REQUIREMENT SPECIFICATION**

**2.1 Problem Statement**

**2.1.1 Bends & Curves**

The current process of managing and designing beam arrangements for structural drawings is complex and time-consuming. Architects, engineers, and designers often face difficulties in visualizing the framework, organizing beam and column details, and ensuring proper beam design for structural strength. Additionally, the lack of a user-friendly interface and automation tools leads to inefficiencies and potential errors in the drawing generation process.

**2.1.2 OfficeMate**

The current administrative process within the organization is manual, time-consuming, and lacks a centralized system for employees to interact with the admin. There is a need for a streamlined solution to simplify administrative tasks and improve communication between employees and the admin.

**2.2 Traditional System**

**2.2.1 Bends & Curves**

Before the advent of specialized software applications, the management and design of beam arrangements for structural drawings were predominantly done manually. Architects, engineers, and designers followed a set of traditional methods to handle these tasks. The traditional method typically involved the following steps:

* + - 1. Manual framework layout: Architects and engineers would manually sketch out the framework of the structure, including the placement of beams and columns. This process required careful measurements and calculations to ensure accuracy.
      2. Paper-based documentation: Beam and column details, such as dimensions, names, and numbering, were recorded on paper. This documentation served as a reference for the construction team and was manually updated as changes were made.
      3. Drawing sheets and templates: Drawings were created on physical drawing sheets using drafting tools like pencils, rulers, and compasses. Standard templates and symbols were used to represent beams, columns, and other structural elements.
      4. Manual numbering: Beams and columns were manually numbered and labeled on the drawing sheets according to a predetermined system. The numbering scheme helped in identifying individual structural elements and their locations.
      5. Manual calculations for beam design: Engineers manually calculated the minimum required steel area (AST) based on the structural specifications and load requirements. Steel bars of appropriate diameters were manually selected and placed in the design to achieve the required strength.
      6. Manual drawing generation: Once the framework, numbering, and beam design were finalized, the construction drawings were manually created on the drawing sheets. This process involved carefully transferring the layout and details onto the sheets, using a combination of freehand drawing and drafting tools.
      7. Manual revisions and updates: As changes or modifications were made to the design, the drawings had to be manually updated. This process often involved erasing and redrawing portions of the drawings, resulting in potential errors and inconsistencies.

The traditional method had several limitations, including the high potential for errors, time-consuming processes, and limited flexibility in making revisions. It required significant manual effort and was prone to inconsistencies, especially when changes were made to the design. The lack of automation and standardized tools made it challenging to efficiently manage and design beam arrangements.

However, advancements in technology have led to the development of specialized software applications that automate and streamline the process, providing a more efficient and accurate solution for managing and designing beam arrangements.

**2.2.2 OfficeMate**

The traditional administrative system within the organization is manual and inefficient, leading to various challenges and drawbacks. The current system lacks automation, centralized data management, and effective communication channels, resulting in time-consuming processes and potential errors.

The identified problems in the traditional system are as follows:

Manual and Paper-based Processes: The traditional system heavily relies on manual

**2.3 Usability Requirements**

The users require Windows operating system.

**2.4 Technologies Used**

**1. C#**

C# is a versatile, object-oriented programming language developed by Microsoft. It is widely used for building a variety of applications, including desktop, web, and mobile applications. C# is part of the .NET framework, which is a comprehensive software platform developed by Microsoft. The .NET framework provides a rich set of libraries, tools, and runtime environments that enable developers to build robust and scalable applications.

One of the key features of C# is its simplicity and readability. It has a syntax that is similar to other popular programming languages such as C and C++, making it easier for developers to learn and write code. C# also incorporates modern programming concepts and features, such as garbage collection for automatic memory management, strong typing, and support for generics. C# supports a wide range of programming paradigms, including object-oriented programming (OOP), functional programming, and asynchronous programming. It includes advanced features like LINQ (Language Integrated Query) that allow developers to perform complex data querying and manipulation operations with ease.

C# is heavily used in the development of Windows desktop applications through technologies like Windows Presentation Foundation (WPF) and Windows Forms. It is also a popular choice for developing web applications using frameworks like ASP.NET. Additionally, C# can be used for cross-platform mobile app development using Xamarin.

Overall, C# provides developers with a powerful and flexible language for building a variety of applications across different platforms, with a focus on productivity, performance, and code maintainability.

**2. .NET**

.NET is a software development framework developed by Microsoft that provides a platform for building a wide range of applications. It encompasses a collection of libraries, tools, and runtime environments that enable developers to create applications for various platforms, including Windows, web, mobile, and cloud.

The key components of the .NET framework include:

1. Common Language Runtime (CLR): The CLR is the execution environment for .NET applications. It provides features like memory management, code execution, security, and exception handling. The CLR enables the execution of multiple languages that target the .NET framework, such as C#, Visual Basic, and F#.
2. Base Class Library (BCL): The BCL is a collection of reusable classes, types, and APIs that developers can utilize when building applications. It provides a wide range of functionality, including data access, networking, file I/O, security, XML manipulation, and more.
3. Language Interoperability: .NET supports multiple programming languages, including C#, Visual Basic, F#, and more. These languages can seamlessly interoperate with each other, allowing developers to utilize existing code and libraries regardless of the language in which they were originally written.
4. Development Tools: Microsoft provides various development tools for building .NET applications, such as Visual Studio, a comprehensive integrated development environment (IDE) with features like code editing, debugging, and project management. Visual Studio Code is another popular lightweight code editor that supports .NET development.
5. Libraries and Frameworks: .NET offers a vast array of libraries and frameworks that extend its capabilities and simplify common development tasks. For example, ASP.NET is a framework for building web applications, Windows Presentation Foundation (WPF) is used for developing desktop applications, and Xamarin enables cross-platform mobile app development.
6. Cross-Platform Development: With the introduction of .NET Core, a cross-platform version of .NET, developers can now build applications that run on different operating systems such as Windows, macOS, and Linux. This provides flexibility and enables the development of cross-platform applications with shared code.

Overall, .NET provides developers with a powerful and versatile framework for building a wide range of applications. It offers a unified development experience, language interoperability, a comprehensive set of libraries, and support for cross-platform development, making it a popular choice for developers worldwide.

**3.WPF (Windows Presentation Foundation)**

Windows Presentation Foundation (WPF) is a graphical user interface (GUI) framework developed by Microsoft as part of the .NET framework. It is specifically designed for creating visually appealing and interactive desktop applications for Windows.

* + - 1. Rich User Interfaces: WPF allows developers to create visually stunning user interfaces with flexible layouts, customizable styles, and a wide range of UI controls. It supports vector graphics, animation, multimedia, and 3D rendering, enabling the creation of visually engaging applications.
      2. XAML for UI Design: WPF uses XAML (eXtensible Application Markup Language) as its declarative markup language for defining the UI. XAML separates the UI design from the application logic, making it easier for designers and developers to collaborate. It provides a clear and concise way to describe the UI elements and their relationships.
      3. Data Binding: WPF offers powerful data binding capabilities, allowing developers to bind UI controls to data sources, such as databases or objects. This enables automatic updating of the UI when the underlying data changes, reducing the need for manual synchronization.
      4. Commanding and MVVM Pattern: WPF includes a commanding system that enables the implementation of common application commands (e.g., copy, paste) without writing boilerplate code. It also encourages the use of the Model-View-ViewModel (MVVM) pattern, which promotes separation of concerns and facilitates testability and maintainability.
      5. Layout and Control Customization: WPF provides flexible layout panels and controls that adapt to different screen sizes and resolutions. Developers can easily create custom controls and styles to match their application's unique look and feel.
      6. Data Visualization: WPF includes powerful data visualization controls, such as charts and graphs, enabling the display of complex data in a visually appealing and interactive manner.
      7. Integration with .NET Ecosystem: WPF seamlessly integrates with other .NET technologies and frameworks, such as the .NET Base Class Library (BCL), LINQ for data querying, and the Entity Framework for data access. It also supports asynchronous programming patterns for responsive and efficient application development.

WPF has been widely adopted for building desktop applications in various industries, including enterprise software, financial applications, media and entertainment, and more. Its flexibility, rich feature set, and XAML-based UI design make it a popular choice for developers seeking to create visually appealing and user-friendly applications for the Windows platform.

**2.5 Implementation Requirements**

**ETABS** - ETABS (Extended 3D Analysis of Building Systems) is a widely used software tool for the structural analysis and design of buildings, providing engineers with advanced capabilities for modeling, analyzing, and simulating the behavior of structures in both static and dynamic conditions.

**Visual Studio Code** - Visual Studio is an integrated development environment (IDE) developed by Microsoft, widely used for building a variety of applications, including desktop, web, mobile, and cloud-based solutions, offering a comprehensive set of tools and features to support the development process.

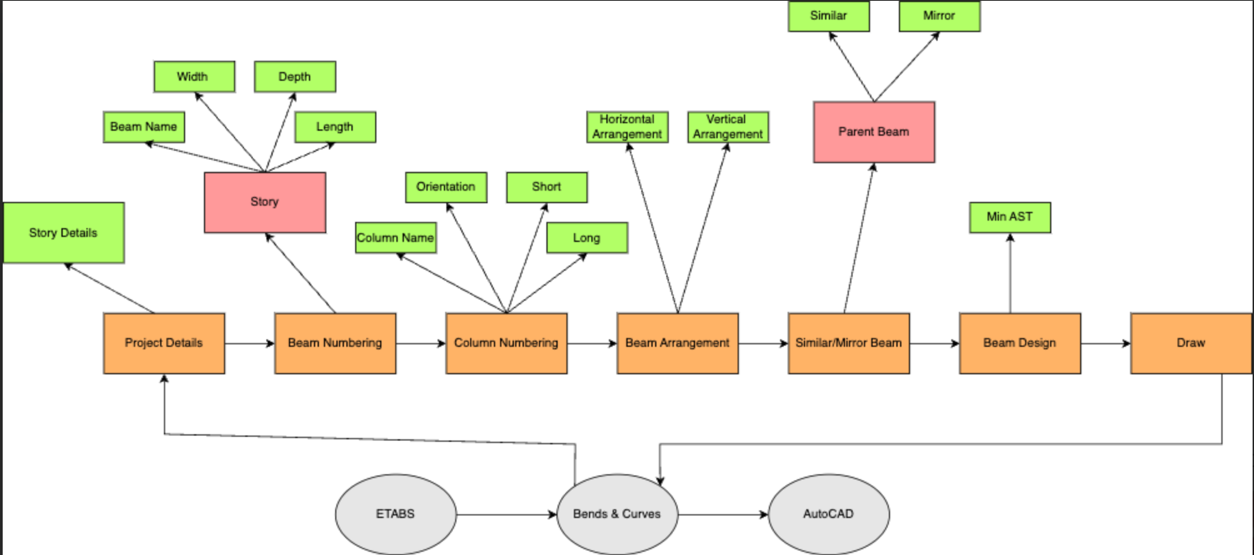
**Microsoft SQL Server** - SQL Server is a relational database management system developed by Microsoft, providing a robust and scalable platform for storing, managing, and retrieving structured data, widely used in enterprise-level applications and data-driven solutions.

**AutoCAD** - AutoCAD is a computer-aided design (CAD) software developed by Autodesk, used for creating precise 2D and 3D models and drawings in various industries, such as architecture, engineering, and construction, with extensive tools and features for design, drafting, and documentation.

**Domain Knowledge** - In order to successfully complete a project, domain knowledge is crucial since it gives the necessary awareness of the procedures, demands, and difficulties unique to a given industry, facilitating efficient decision-making and solution design.

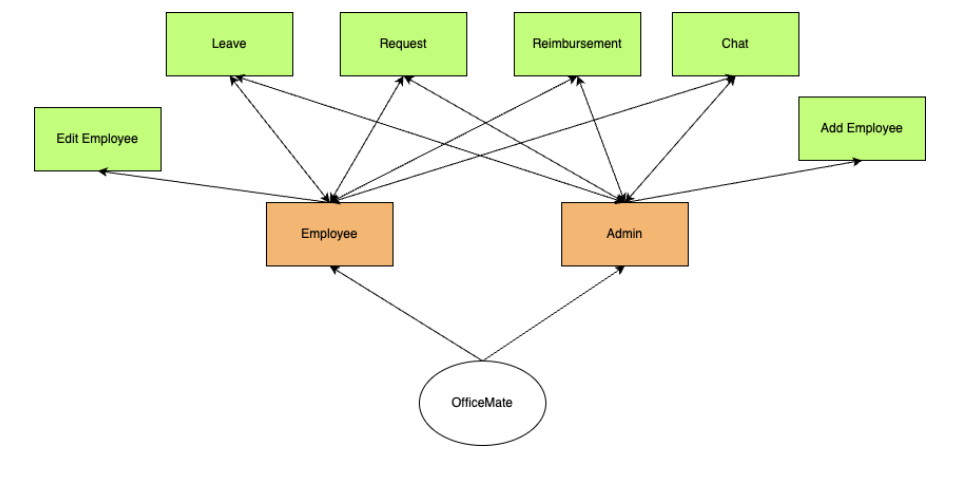
**2.6 Data Flow Diagram**

**2.6.1 Bends & Curves**



2.1 Flow of Bends & Curves

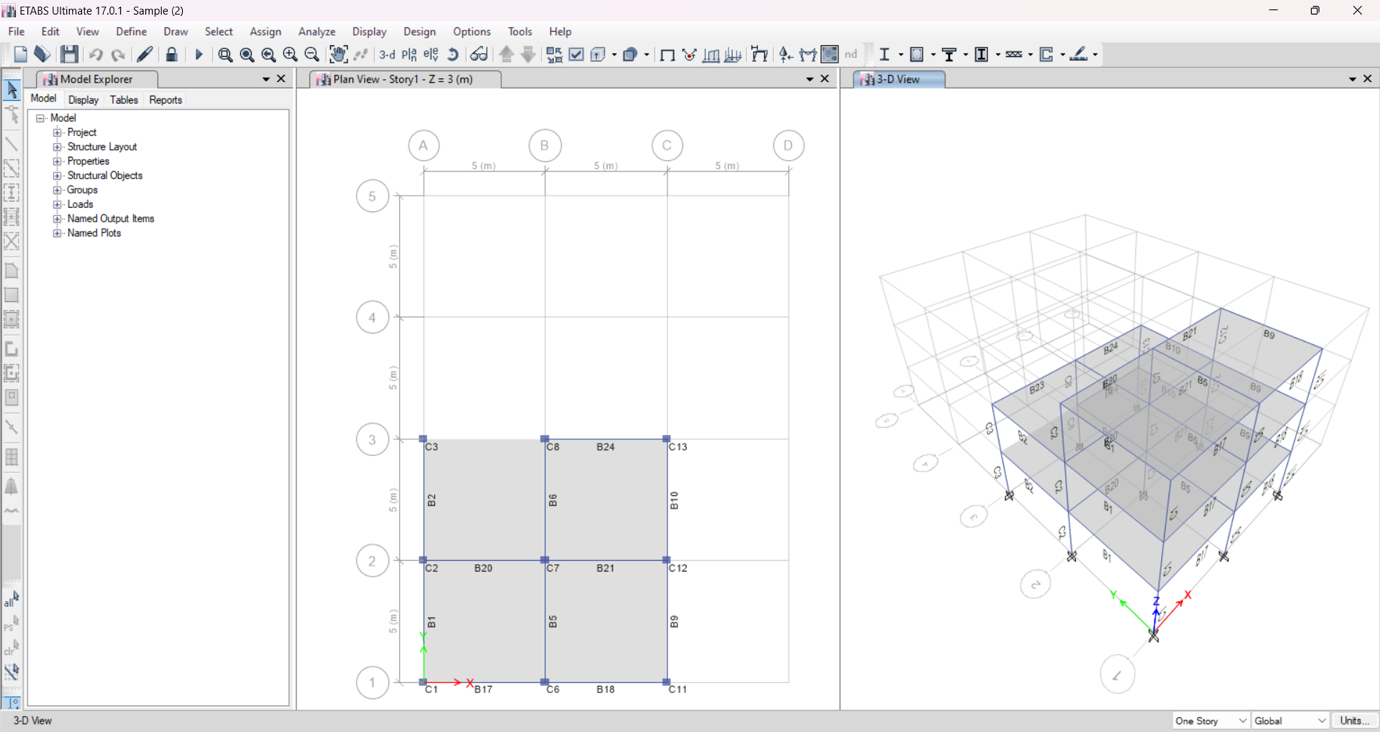
**2.6.2 OfficeMate**



2.2 Flow Diagram of Officemate

**IMPLEMENTATION**

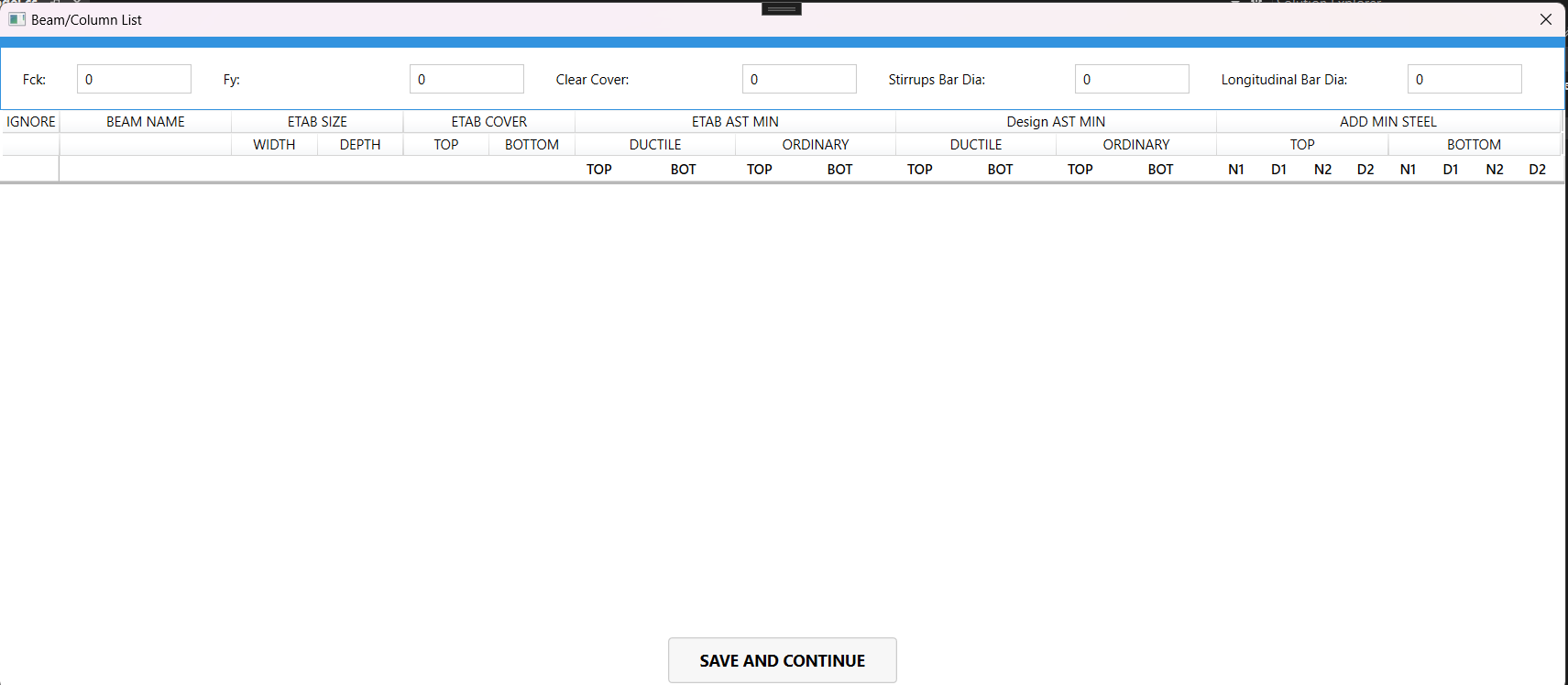
* 1. **Bends & Curves**
     1. **ETABS Window**

****

3.1 ETABS

Design is drawn in ETABS and saved in .ebk format.

**3.1.2 Beam Column List**

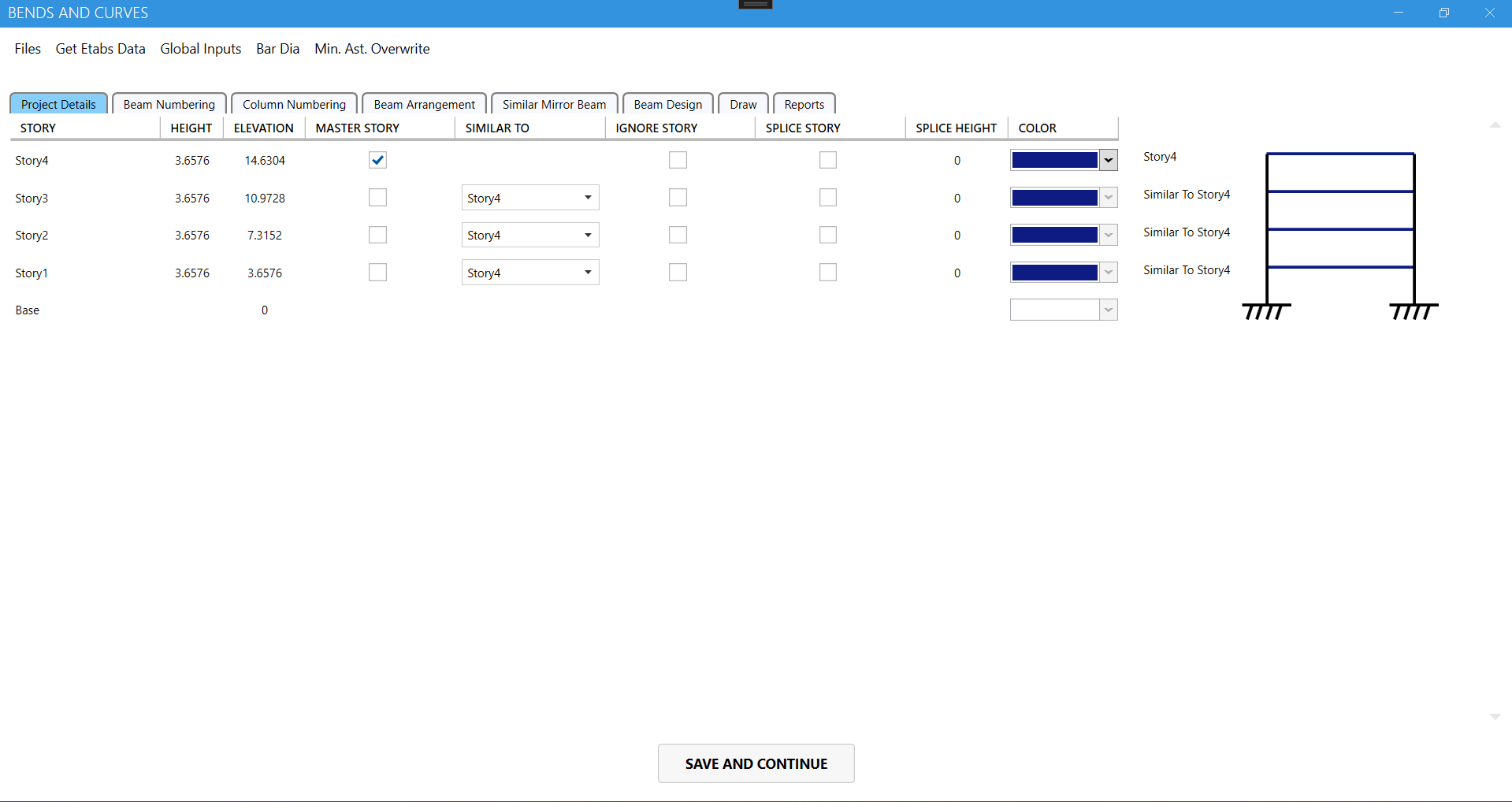
****

3.2 Beam Column List

All data is fetched from ETABS using ETABS API. All columns and beams will be displayed here.

**3.2.3 Main Application Window**

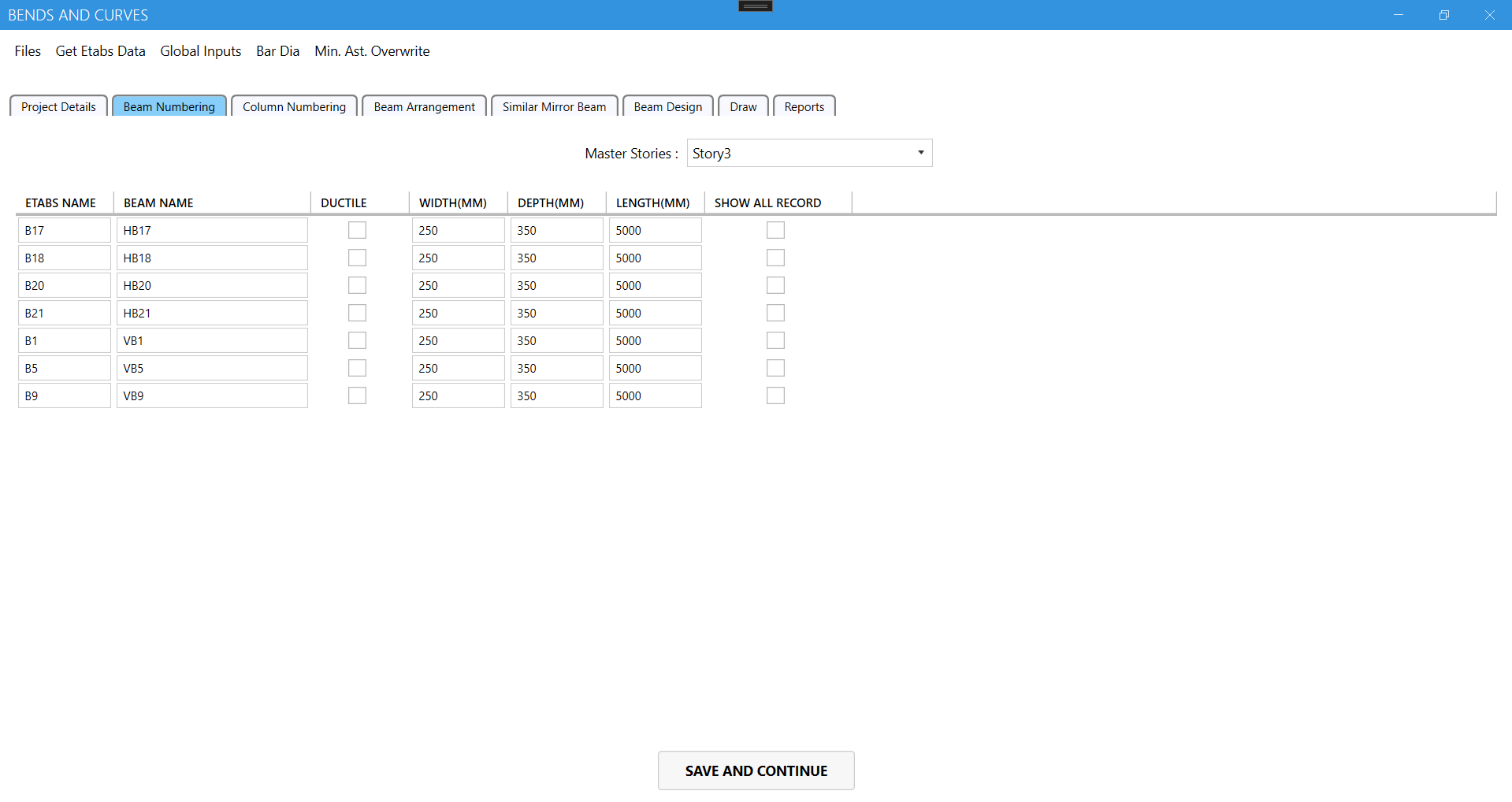
1. **Project Details**

****

3.3 Project Details

In this tab we can see the stories in the drawing draw on ETABS. We aan set the master story and set any story similar to other story are ignore a specific story .

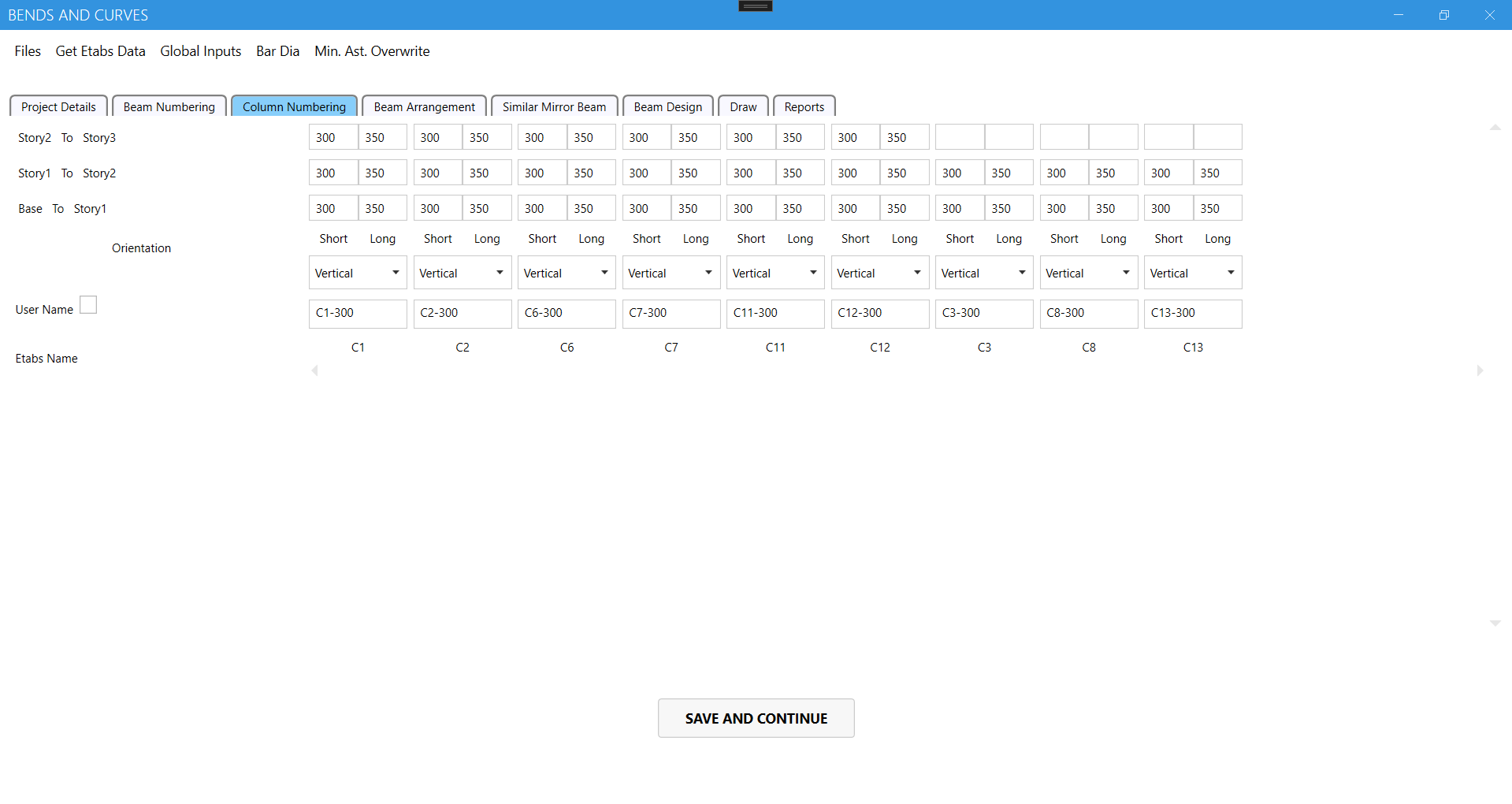
**2. Beam Numbering**

****

3.4 Beam Numbering

In Beam numbering tab we can see the all the beams in the drawing with etabs name and the custom name given by us and width, depth and the length of the beam. We can change it from here easily.

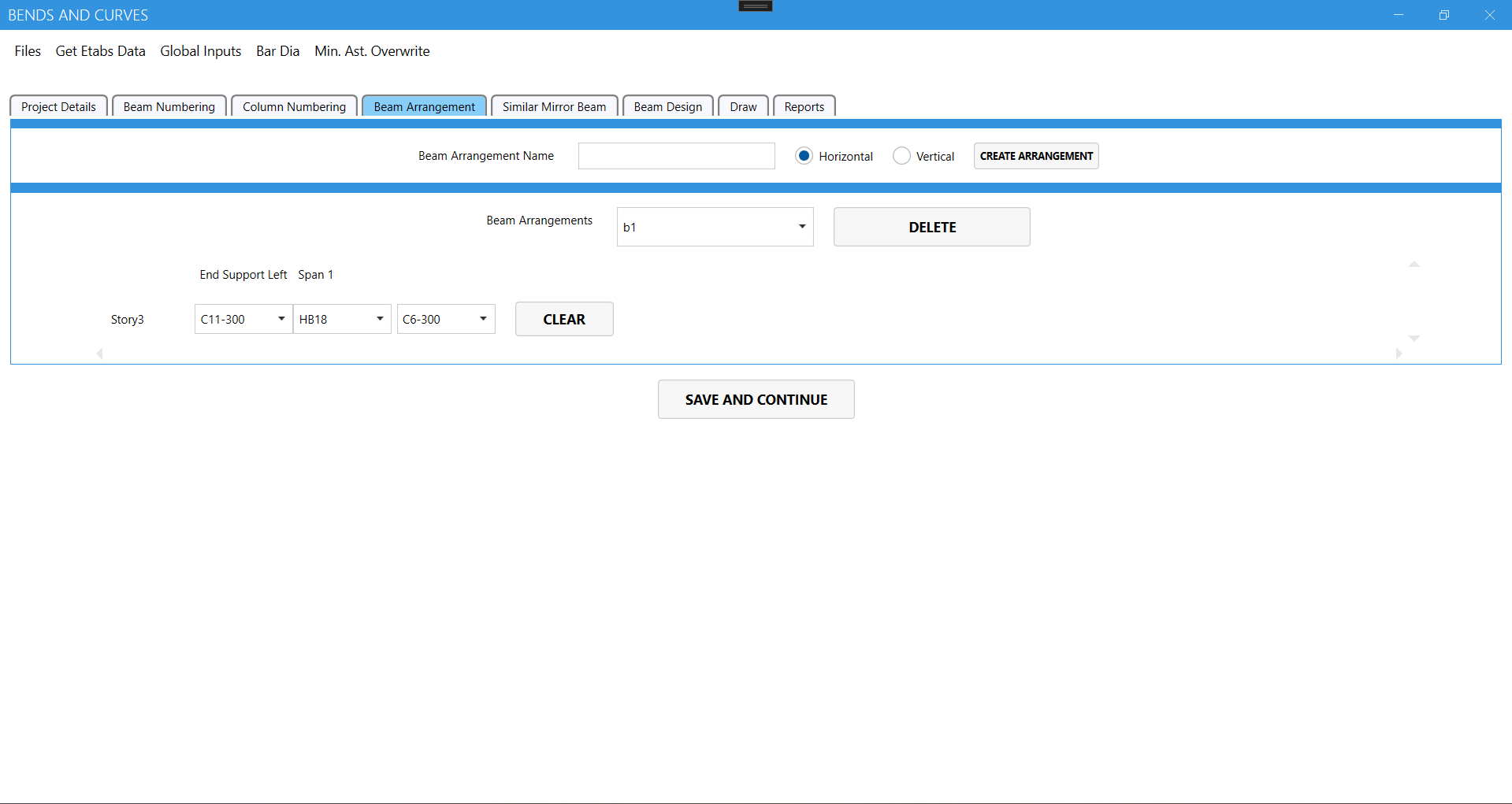
**3. Column Numbering**

****

3.5 Column Numbering

Here you will get the colums story wise with their dimensions and the orientation and the ETABS name and custom name same as the beam numbering.

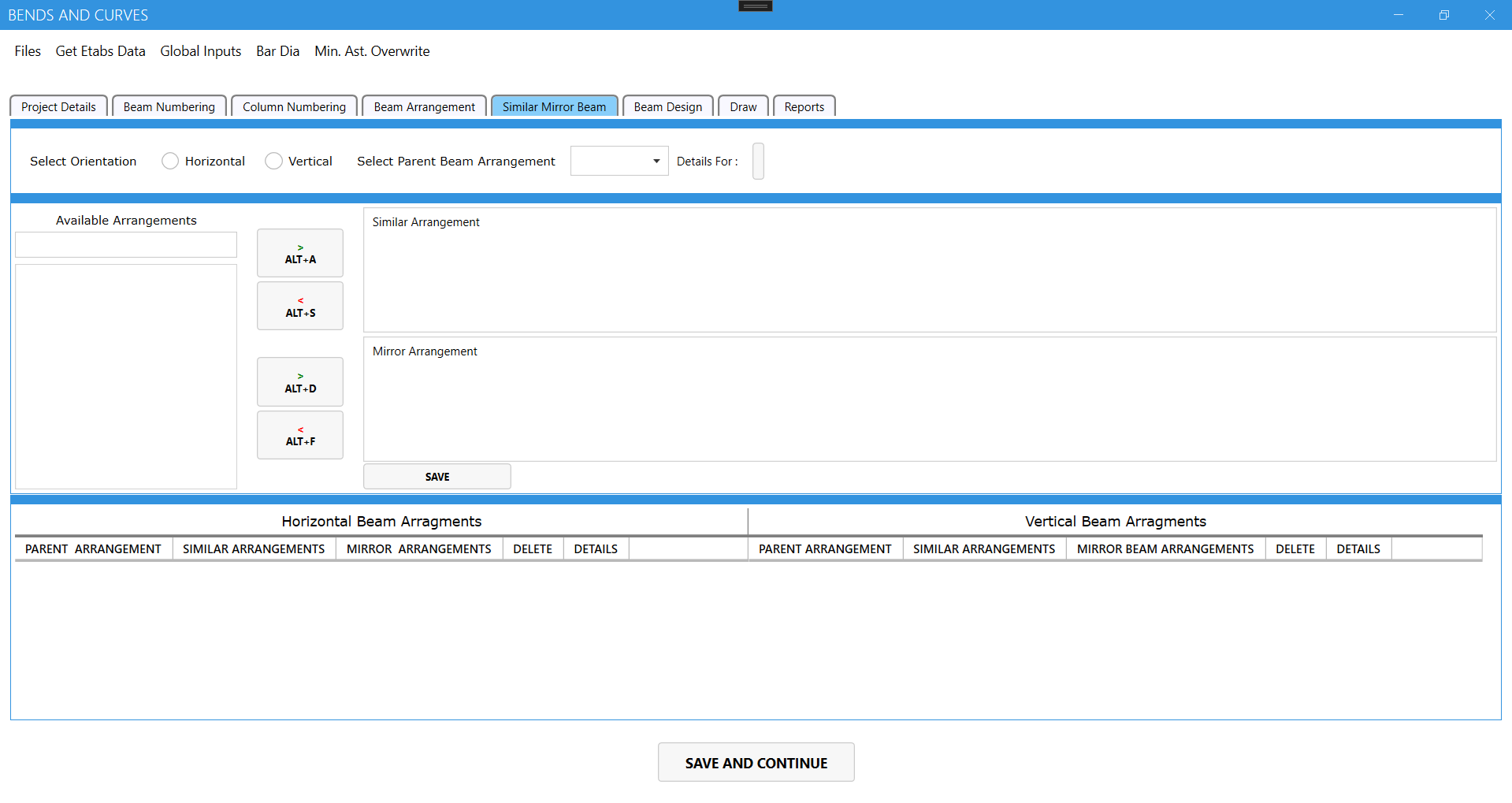
**4. Beam Arrangement**

****

3.6 Beam Arrangement

In beam arrangement we can create various arrangements with different orientations as per requirement and save them.

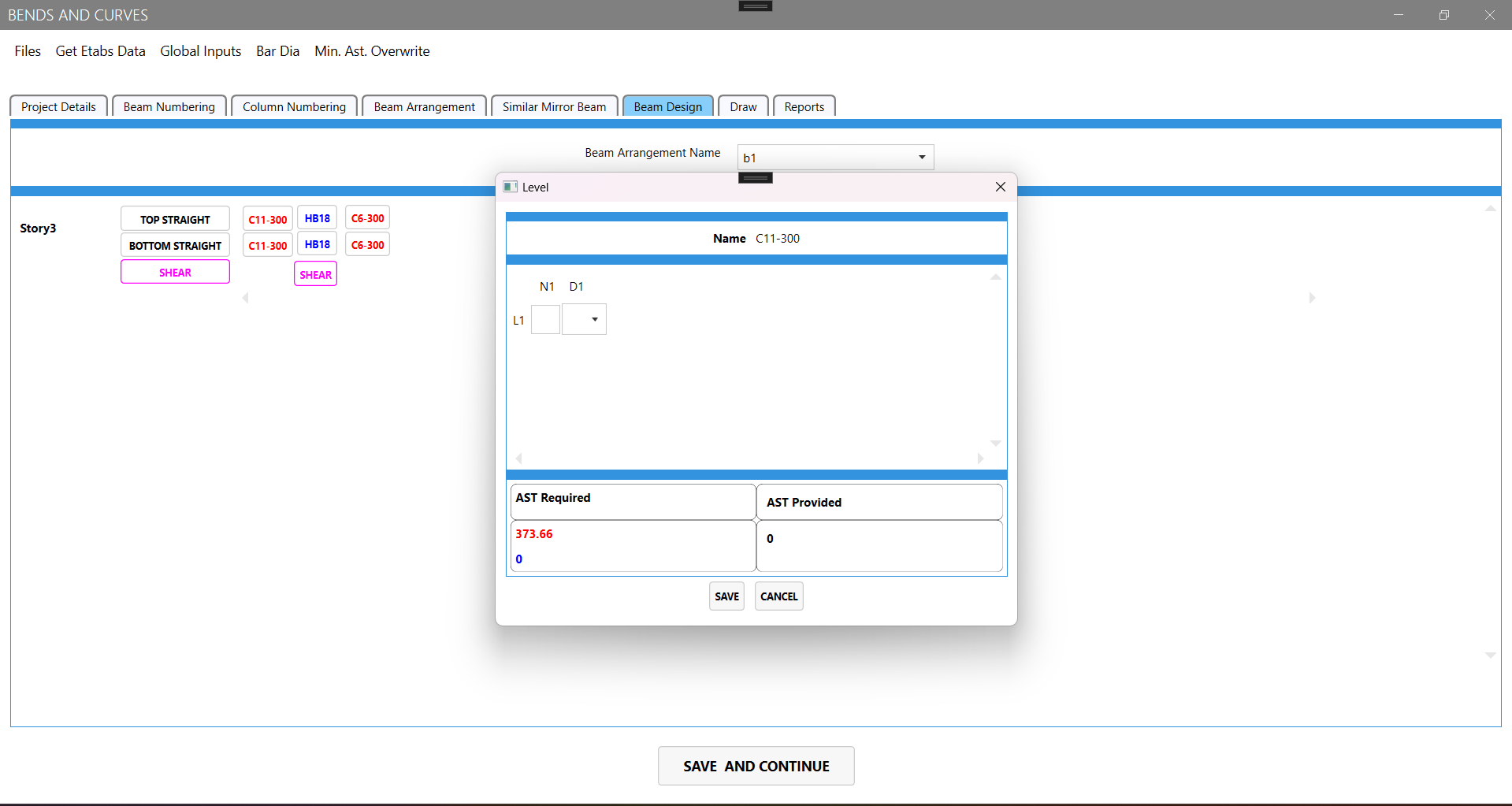
**5. Similar/Mirror Beam Arrangement**

****

3.7 Similar / Mirror Beam Arrangement

In this tab we can select a parent beam arrangement and set similar and mirror according to the requirement.

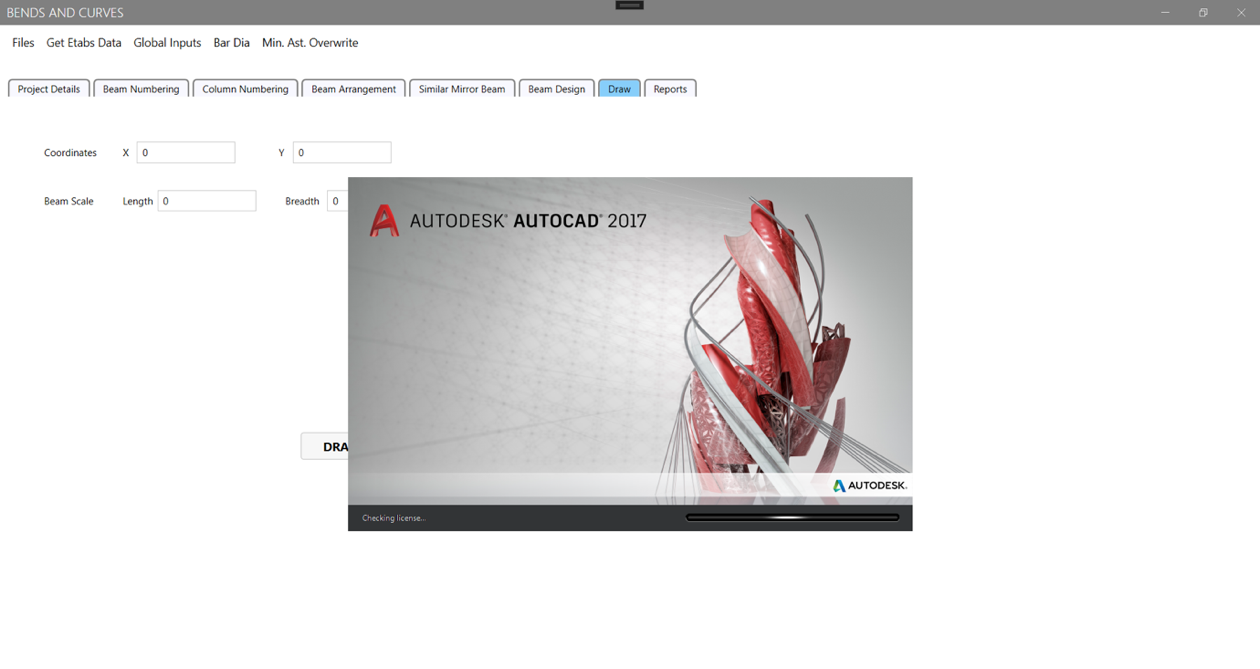
**Beam Design**

****

3.8 Beam Design

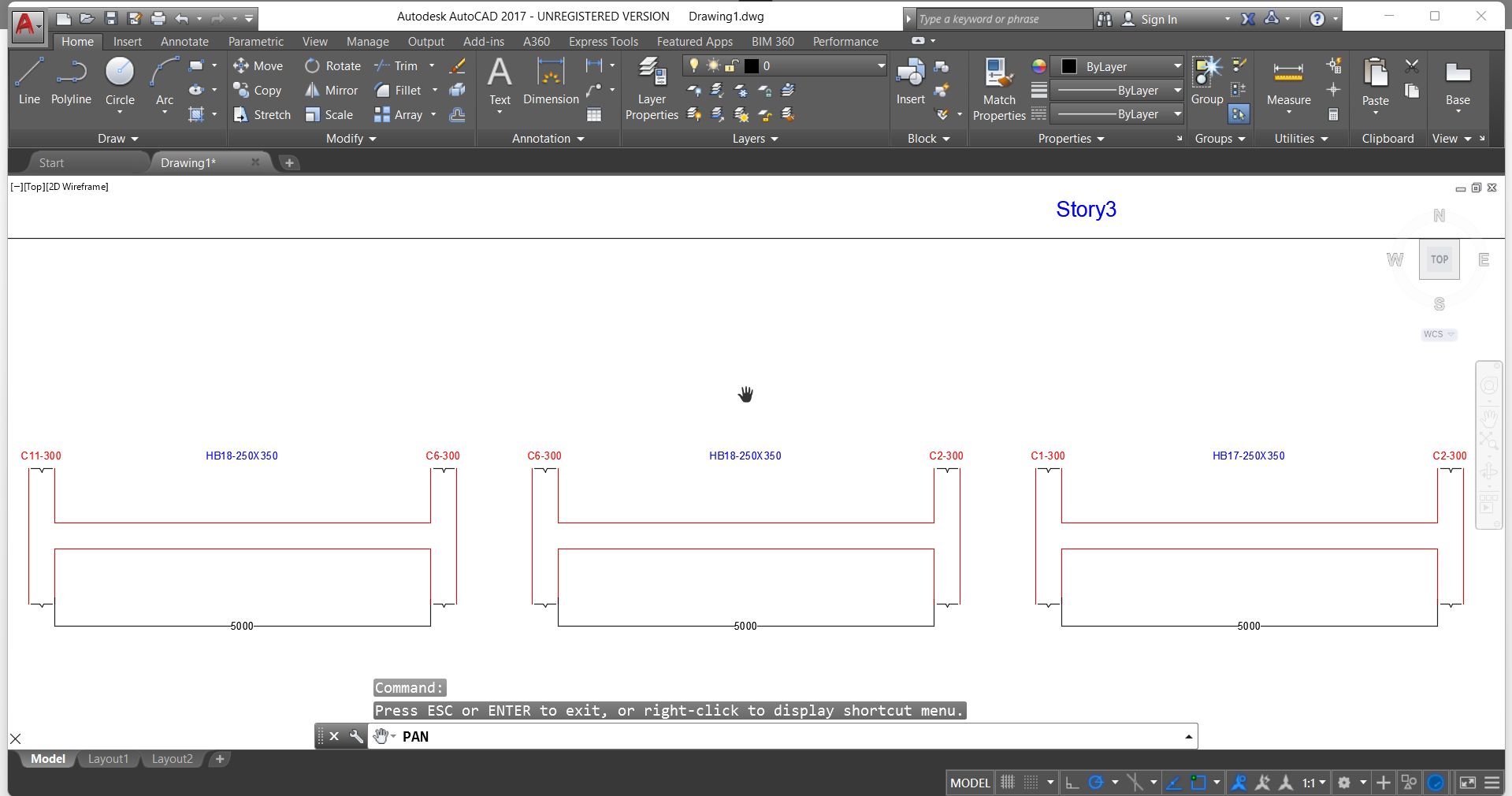
In Beam design tab we can see the min AST required asper that we can add the steel of different diameters and fulfil the min AST.

**Draw**

****

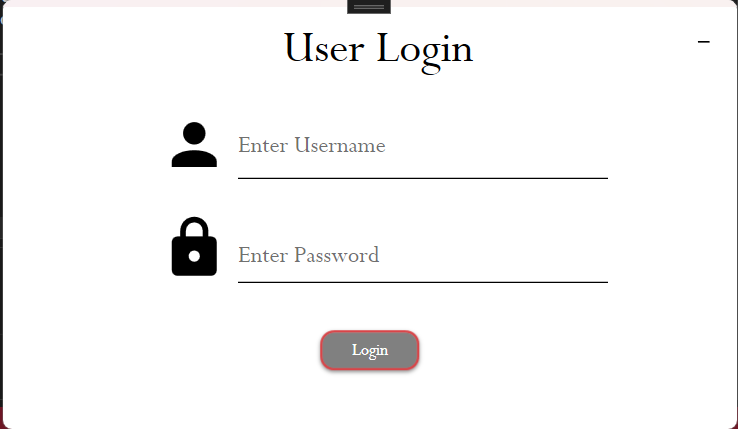
3.9 Draw

When we hit draw button in draw tab AutoCAD fires automatically and the drawing is drawn accordingly which is designed in the Application in the beam arrangement tab. Drawings are drawn story wise in different rectangular sheet.



3.10 AutoCAD

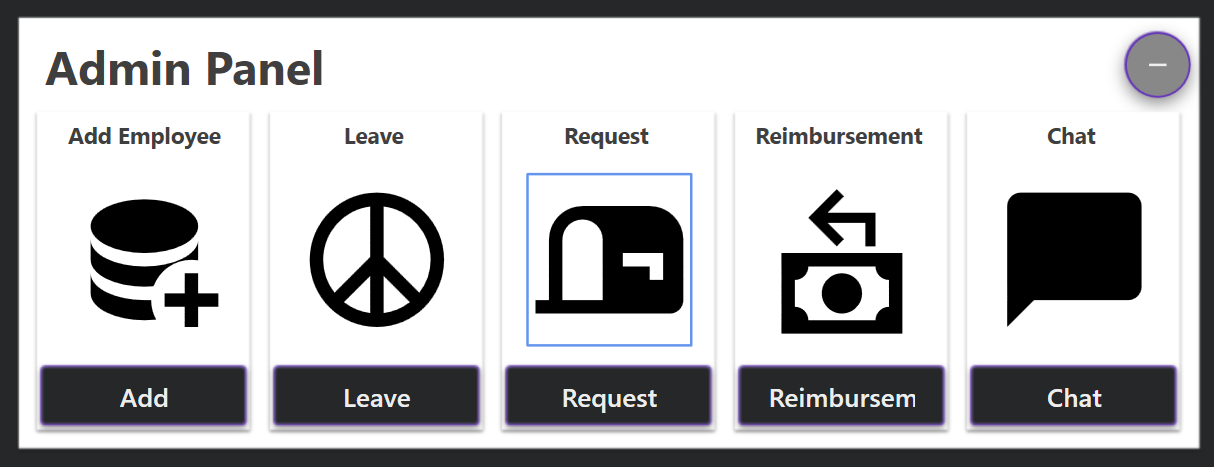
* 1. **OfficeMate**
     1. **Login Page**



3.11 Login Page

This is the login page is same for Employee and Admin. Depending on the role given to the credentials admin panel and the employee panel opens.

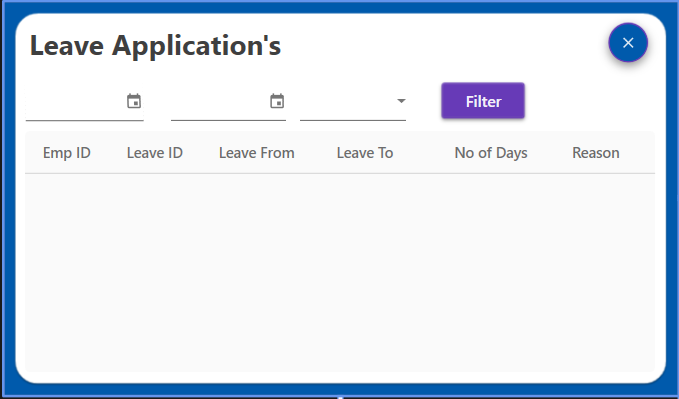
**3.1.2 Admin Panel**

****

3.12. Admin Panel

In admin Panel we have five main functions that are mentioned below

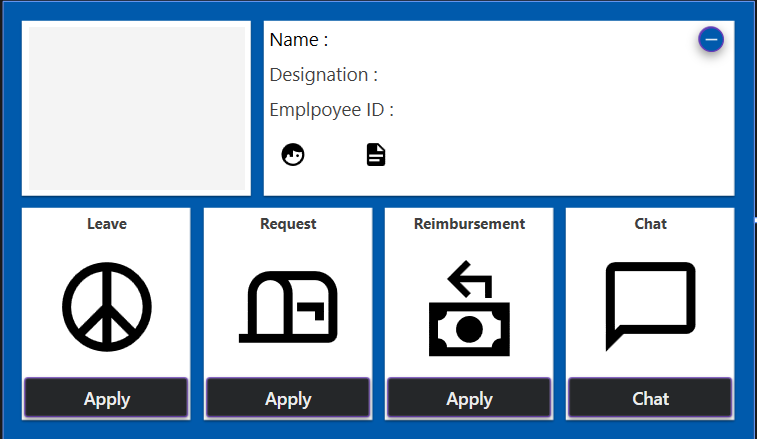
1. Add employee : Whenever new employee joins admin can add employee to the application with new username and password.
2. Leave : Whenever the employee applies for the leave admin can review it and accept or reject accordingly.
3. Request: Whenever the employee applies for the request admin can review it and accept or reject accordingly.
4. Reimbursement : Whenever the employee applies for the reimbursement admin can review it and accept or reject accordingly.
5. Chat : Admin can chat with all the employees through this application only in this network.
6. **Leave Application**

****

3.13 Leave Application

When the employee applies for leave that leave application will be visible to admin in this tab this tab has several details like Emp ID , Leave ID , Leave from/to Dates, No of days and reason. Same goes with the Reimbursement and Request applications.

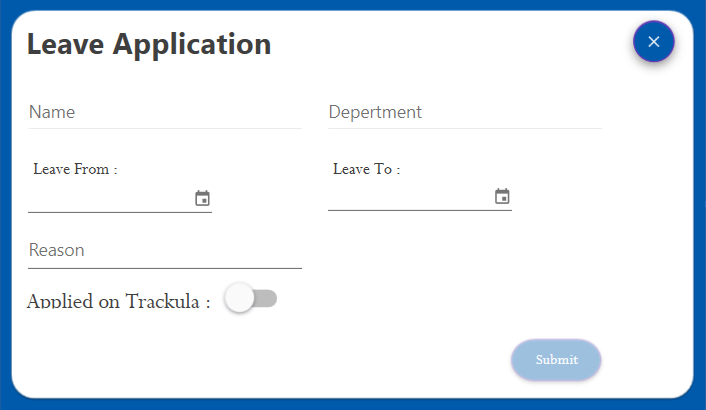
**3.1.3 Employee Panel**

****

3.14 Employee Panel

In employee panel at top right side card you will see the employee details with edit profile and the report button at the left top corner you will see the profile picture are when employee updates the profile picture it will be displayed there at bottom we have four cards they can be used for applying leave, request, reimbursement, chat.

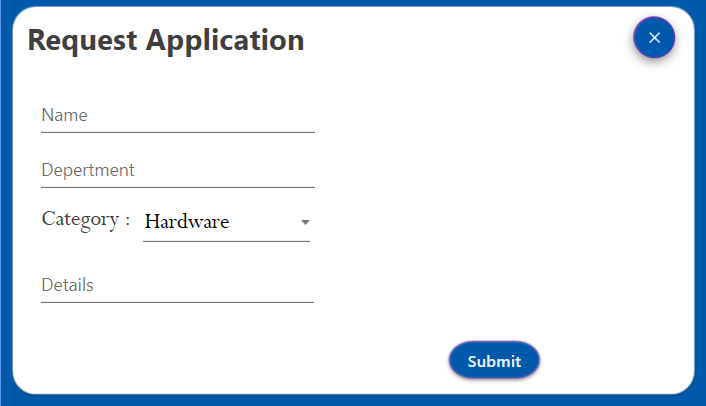
1. **Leave Application**



3.15 Leave Application

In leave application employee has to mention all details all details are required to submit the application and the application must be also applied on a software called trackula so it can be reviewed properly and then accepted.

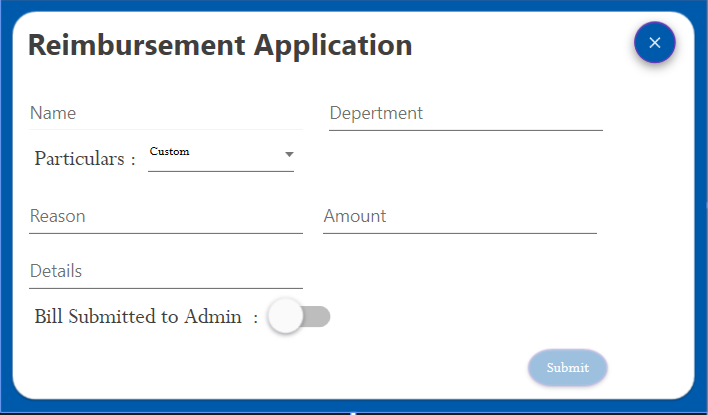
1. **Request Application**



3.16 Request. Application

Request tab only has to mention Category and details. Name and department will be retrived from database automatically as per the credentials.

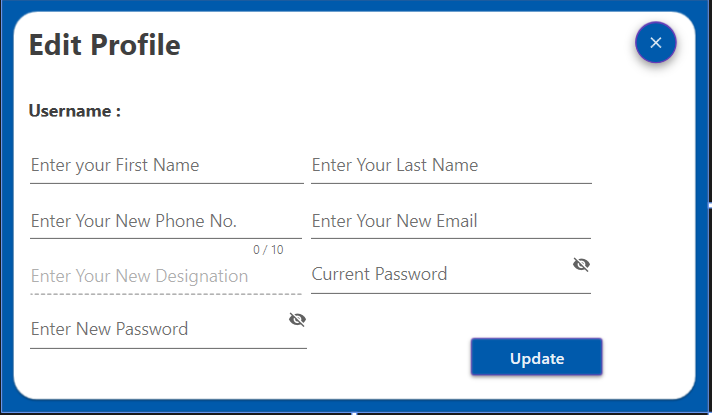
1. **Reimbursement Application**



3.17 Reimbursement Application

On this page all fields are required and particular has various common things or a other option amount and details and can be applied only when the bill is already submitted to admin.

1. **Edit Profile**



3.18 Edit Profile

In edit profile tab designation and username is not editable and all other fields are editable and password will be updated only when old password matches with the database. All fields have validations.

**4. Result and Discussion**

1. **Bends & Curves**

Bends & Curves provides a simple and user-friendly dashboard for creating and managing drawings. It focuses on displaying the basic framework of the drawing, organizing stories, identifying similar stories, and determining master stories

Dashboard: The application offers a dashboard that provides an overview of the drawing and its stories. It presents information about the relationships between stories, such as which ones are similar to each other and which ones are ignored. Additionally, it identifies any master story that may exist.

Overall, this drawing application provides a comprehensive set of tools for creating, organizing, and designing drawings, making it easier for users to visualize and manage structural elements.

1. **OfficeMate**

OfficeMate is a Windows application designed to simplify administrative tasks and facilitate communication between employees and the admin. It includes the following modules:

The OfficeMate application aims to streamline administrative tasks and enhance communication between employees and the admin, improving overall efficiency and productivity in the workplace.

1. **Conclusion**

Both projects, the Bends & Curves and OfficeMate, aim to improve efficiency and organization within their respective domains.

The Bends & Curves provides a user-friendly dashboard for creating and managing drawings. It offers features such as beam and column numbering, beam arrangement, beam design, and automatic drawing generation. This application streamlines the process of designing structural elements and helps users visualize and manage their drawings effectively.

On the other hand, OfficeMate is a Windows application designed to simplify administrative tasks and facilitate communication between employees and the admin. It includes modules for leave applications, request applications, reimbursement applications, and a chat feature. This application enhances the efficiency of administrative processes and provides a convenient platform for employee-admin communication.

Both projects demonstrate a focus on usability and streamlining processes. They offer intuitive interfaces and features that cater to the specific needs of their respective domains. By leveraging technology, these applications aim to optimize workflow, improve productivity, and enhance communication within their target environments.

In conclusion, the Bends & Curves and OfficeMate contribute to increasing efficiency and organization in their respective fields. They provide practical solutions that simplify complex tasks, streamline processes, and foster effective communication. Implementing these applications can lead to significant improvements in productivity and overall operational effectiveness.

1. **Future Scope**

Both the Bends & Curves and OfficeMate projects have potential for future development and expansion. Here are some possible areas of future scope for each project:

**Bends & Curves**

1. Advanced Design Capabilities: The application can be further enhanced with advanced design capabilities, such as 3D modeling, simulations, and structural analysis. This would enable users to visualize and analyze their designs in more depth.
2. Collaboration Features: Introducing collaboration features would allow multiple users to work on the same drawing simultaneously, facilitating teamwork and coordination. Real-time updates, version control, and commenting functionalities could be implemented to streamline collaboration.
3. Integration with BIM Tools: Building Information Modeling (BIM) is a widely used methodology in the construction industry. Integrating the drawing application with BIM tools would enable seamless data exchange, enhancing interoperability and improving the overall design and construction process.
4. Mobile Application: Developing a mobile version of the drawing application would provide users with the flexibility to create and manage drawings on their smartphones or tablets. This would be especially beneficial for field engineers and architects who require access to drawings on-site.

**OfficeMate:**

1. Workflow Automation: Implementing workflow automation features would further streamline administrative processes. For example, incorporating approval workflows for leave applications, reimbursement requests, and other employee requests could reduce manual intervention and expedite decision-making.
2. Employee Self-Service Portal: Introducing a self-service portal for employees would empower them to access and manage their own information. They could view their leave balances, track reimbursement statuses, and update personal details, reducing the administrative burden on HR personnel.
3. Reporting and Analytics: Adding reporting and analytics capabilities to OfficeMate would provide valuable insights into employee trends, leave patterns, and reimbursement expenses. This data could help organizations make informed decisions and identify areas for improvement.
4. Integration with HR Systems: Integrating OfficeMate with existing HR systems, such as payroll and attendance management, would create a unified ecosystem. This integration would ensure accurate data synchronization and minimize duplication of effort.
5. Expansion to Other Platforms: Besides Windows, expanding OfficeMate to other platforms, such as web-based applications and mobile apps (iOS and Android), would increase its accessibility and user base.

By exploring these future scope opportunities, both projects can continue to evolve, meet evolving user needs, and stay relevant in their respective domains.

1. **References**
2. **C#WPF-**

<https://youtu.be/Vjldip84CXQ>

<https://youtu.be/FGqj4q09NtA>

https://youtu.be/zvyQNuuTqks

1. **SQL-**

https://www.w3schools.com/sql/

https://youtu.be/EKOfCbxt5Po