# Decidio 2.0: Enabling Enhanced Collaborative Decision-Support and Interaction

We present Decidio 2.0, an enhanced iteration of our previously introduced decision-support system. The original Decidio offered a simple framework in which researchers could configure a variety of structured interactions ("meetings") in order to experiment with the effect of rules and machine presentation on meeting outcomes. Building upon the foundations laid in the Decidio study, Decidio 2.0 implements a richer control model for interaction, and thus offers researchers more nuanced behavior in the building blocks available to them in their studies. The enhanced configurability extends the dynamic updates in breakout rooms, video conferencing and harvesting of user feedback. Virtual participants ("bots") are enabled to support experiments with group dynamics; we offer a simple API which allows researchers to focus on experimenting with their own machine learning ideas while the framework handles the rest. Decidio 2.0 also incorporates an interface to dovetail the decision capabilities with custom tools for managing content and curating participant reputation, which we introduce in a new research project called Colloquium. Through a comprehensive evaluation and user study, we assess the impact of these enhancements on decision-making processes, team dynamics, and user satisfaction. Our findings contribute to the evolving landscape of decision-support systems, offering valuable insights into the potential of collaborative technology to facilitate more informed and efficient decision-making in various settings.

## 1  INTRODUCTION

An interactive collaborative platform is a type of software or digital tool that enables individuals or groups to work together on a shared task or project in real-time. These platforms [11] typically provide a variety of features and tools designed to facilitate collaboration and communication, such as chat functions, video conferencing, shared file editing, and project management tools. Interactive collaborative platforms are used in a wide range of contexts, from remote work and online learning to virtual brainstorming sessions and team projects. They can help to overcome geographical barriers and enable people to work together from anywhere in the world.

In recent years, the use of interactive collaborative platforms [6, 12, 14, 15] has become increasingly common due to the rise of remote work and the growing need for flexible and agile ways of working. These platforms can help to boost productivity, enhance creativity and innovation, and foster a sense of connection and community among team members, even when they are physically located in different parts of the world.

A design support system (DSS) is defined as "interactive computer-based systems, which help decision makers utilize data and models to solve unstructured problems" [8]. Decidio [10] is a pilot DSS and a web-based collaborative ranking platform to support groups that need to collectively rank a set of options, which has been trialed in a user study, to test its effectiveness. In this work we introduce Decidio 2.0, an improved version based on Decidio's study which offers integration with external platforms and APIs, advanced meeting controls, decision-based breakout room navigation, video conferencing and automated user-feedback.

We also introduce Colloquium, a robust peer review system, where scholars can experiment with methods to evaluate, critique, and provide constructive feedback on one other's work, thereby contributing to the quality and rigor of academic output. Colloquium offers a work rating mechanism that allows participants to rate and rank the quality and relevance of research contributions, providing a valuable measure of the significance of different works within the academic community. It also features a participant reputation system that helps recognize and acknowledge the expertise and contributions of individuals involved in the platform. Reputation systems serve as a testament to the experience, knowledge, and active engagement of participants, fostering a sense of trust and accountability in collaborative academic environments. These systems are natural partners and this paper illustrates the merits of integrating structured meeting services with rich applications. In this case the reviews

that might be conducted in Decidio 2.0's meetings can be curated in Colloquium, and those data in turn inform other meeting types so later participants can better weigh the input. These labels thus enable experiments with machine learning techniques to close the loop between reviews and outcomes, or said another way, in future discussions we can start to learn what properties to look for as markers of best advice.

## 1.1 Current state of interactive collaboration platforms

The current state of interactive collaboration platforms is quite advanced and constantly evolving [4, 11]. With the increasing demand for remote work and distributed teams, there has been a significant growth in the number and variety of collaboration tools and platforms available.

Many of the existing platforms [11] offer a wide range of features designed to enhance collaboration and productivity, including real-time document editing, video conferencing, project management tools, and chat functions. Some platforms are geared towards specific industries or use cases, such as virtual event platforms, design collaboration tools, or project management software.

Additionally, many collaboration platforms have also incorporated artificial intelligence and machine learning technologies to improve the user experience and streamline workflows. For example, some platforms can automate repetitive tasks, suggest meeting times, or provide personalized recommendations based on the user's behavior.

Overall, the current state of interactive collaboration platforms is very promising, with many tools and technologies available to help people work together more effectively and efficiently, regardless of their location.

## 1.2 What is missing from the current platforms?

Although many current collaborative tools support audio/video and screen sharing, they often lack the capability for developers to plug-n-play external platforms and APIs to better facilitate decision-making processes. They also lack the capability to create decision-oriented breakout rooms to facilitate discussion among participants and integration of bots into the discussion to help users better participate in a meeting.

Decidio 2.0 is created to address the above gaps in collaborative decision-making. Our platform enables integration of 3rd party platforms, APIs and bots to utilise Decidio 2.0's decision making process. We also introduce decision-guided breakout-rooms creation on the fly to allow participants to reach a consensus with varying opinions on multiple topics. A primary objective has been to decrease the expenses associated with testing new user experiences and rules. This enables us to replicate what others have created while also allowing individuals to experiment with different variations inexpensively, which we will assist through demonstrations. Additionally, we believe that facilitating meetings is comparable to managing workflow in an office, and thus we have incorporated features to support this. We also introduce Colloquium an external platform built to showcase the integration of Decidio 2.0 interfaces.

## 1.3 What could be the advantages/benefits if we integrate the missing components?

There are many potential benefits of an interactive platform with such features. They include:

(1) **Cost Reduction**: Decidio 2.0 reduces research costs by integrating third-party tools, allowing cost-effective experimentation, and facilitating consensus among researchers. This platform streamlines processes, minimizes meeting-related expenses, and ultimately optimizing the return on investment for research endeavors.
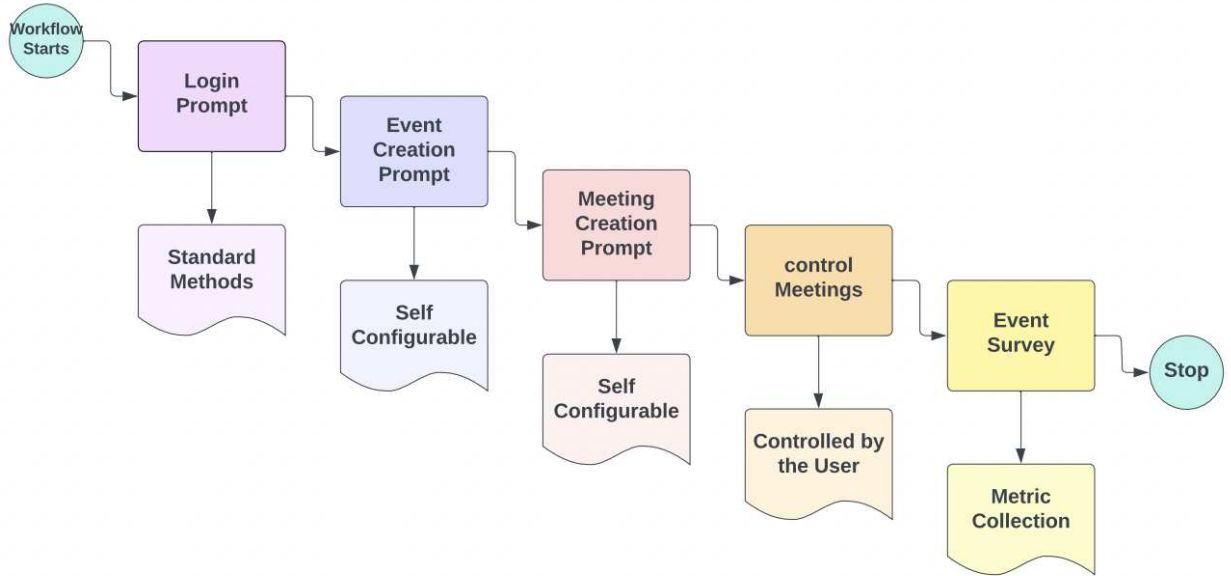
Fig. 1. Functional Flow Diagram

(2) **Recommendations**: We leverage user metrics to generate recommendations aimed at enhancing the user experience. These insights allow the platform to proactively suggest tips, hints, and personalized content to users.

(3) **External Platform Integration**: External platforms like Colloquium can integrate with Decidio 2.0 to curate advice and facilitates machine learning experiments for improved review outcomes.

(4) **Better decision-making**: By enabling decision guided breakout rooms, group-input sessions and poll-meetings, the platform can help facilitate better decision-making among group of users.

(5) **Improved Workflow**: The platform can help save time by enabling users to conduct meetings remotely and access important features within the application, rather than having to use multiple tools.

(6) **Improved collaboration**: With the ability to set up different types of meetings and access a wide range of in-application features, users can collaborate more effectively and efficiently.

(7) **Increased engagement**: Interactive meetings with audio/ video and remote-desktop features can help to increase engagement and participation among users.

(8) **Enhanced productivity**: With a platform that is designed for efficient collaboration, teams can work more productively and complete tasks more quickly.

In summary, an interactive platform like Decidio 2.0 has the potential to improve communication, collaboration, and negotiation, leading to more successful outcomes for teams and organizations.

## 1.4 Examples of some scenarios that could benefit from our interactive platform

Here are a few examples of scenarios that could be benefited by using the interactive platform Decidio 2.0:

(1) **Focus Groups**: Decidio 2.0 offers a dynamic forum for participants to engage in in-depth discussions about their experiences with the platform. On-the-fly creation of decision-guided breakout rooms, facilitate discussions and consensus-building among diverse stakeholders with varying opinions.

(2) **Project Brainstorming**: A team of researchers from different departments needs to collectively brainstorm and prioritize ideas for an upcoming project. They can use Decidio 2.0 to create a decision-guided breakout room where they can discuss and evaluate each idea, reaching a consensus efficiently. They integrate external research data via APIs to inform their decisions.

(3) **HR and Recruitment**: The HR department can leverage Decidio 2.0 to manage the recruitment process more efficiently. They integrate job portals and use the platform's features to coordinate interviews and assess candidate feedback collaboratively.

(4) **Marketing Campaign Planning**: A marketing team can use Decidio 2.0 to plan a new advertising campaign. They create a decision-guided breakout room to discuss various aspects of the campaign, including target demographics, messaging, and budget allocation. By involving team members with diverse expertise, they create a comprehensive and effective campaign strategy.

(5) **Project Management**: Decidio 2.0 can be used for project management where project managers can hold review meetings and track the progress of their projects. The in-application features such as task management, document sharing, and real-time editing can streamline project management and improve team collaboration.

(6) **Customer Engagement**: Decidio 2.0 can be used for customer engagement where companies can hold interactive meetings with their customers, gather feedback, and collaborate on product development. The platform can also be used for customer support, allowing agents to engage with customers in a more personalized and interactive way.

(7) **Online Education**: Decidio 2.0 can be used for online education where instructors can create interactive sessions that allow students to collaborate, participate in polls, and engage in group activities. The platform can also be used for remote tutoring and one-on-one sessions.

(8) **Team Meetings**: Decidio 2.0 can be used for team meetings where team members can collaborate and share their ideas using group input, poll meetings, and review meetings with audio/video on. The in-application features such as chat, file sharing, and real-time editing can enhance the collaboration experience.

(9) **Remote Work**: With the rise of remote work, Decidio 2.0 can be a useful tool to bring remote team members together in a virtual space. The audio/video conferencing features can enable face-to-face communication and make remote work more engaging.

## 2 IMPLEMENTATION

### 2.1 Functional Overview

The workflow within Decidio 2.0 closely mirrors the depiction in Figure 1. To engage with the system, any user is presented with the choice of logging in through established third-party applications such as Google or Facebook, or alternatively, by registering as a new user for seamless access.

Upon successful login, the gateway to system interaction unfolds with the Event Creation/Update Prompt. Within this pivotal feature set, users find a spectrum of commanding capabilities at their fingertips. This includes the ability to not only initiate and craft new events but also to breathe fresh life into existing ones.

Furthermore, users can deftly oversee meetings, exercising precise control to ensure seamless and productive interactions. In addition to its organizational prowess, the platform invites users to partake in its continual

evolution through feedback surveys, fostering a dynamic ecosystem of enhancements and refinements. These collective functionalities empower users to effortlessly navigate, orchestrate, and optimize their involvement in the platform's events and meetings, enriching their overall experience.

In the upcoming sections, we'll delve into the core concepts driving Decidio 2.0's creation. We'll explore its components, including Breakout Rooms, Event Monitoring, Event Interaction Metrics, and Interaction Atomic Units. Additionally, we'll dissect its architectural framework, demonstrating the integration of third-party platforms. Finally, we'll introduce Colloquium, an external platform that showcases Decidio 2.0's interfaces and its commitment to improving collaborative decision-making. These insights will paint a comprehensive picture of Decidio 2.0's innovative approach and its role in revolutionizing collaborative decision processes.

## 2.2 Meeting as a Service

We introduce Meetings-as-a-service, by defining meeting as a big component comprising of 3 sub components. The following Figure 2 clearly illustrates the sub-parts.
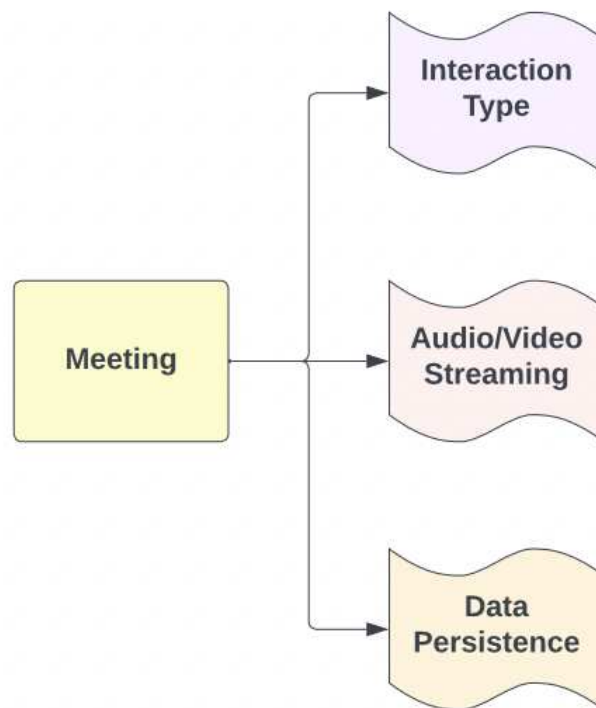


Fig. 2. Meeting Components

(1) **Audio/Video Streaming**: Audio and video streaming hold a pivotal role in contemporary meetings, offering a multifaceted array of advantages that elevate the quality and efficacy of interactions. They

provide real-time, synchronous communication, bridging geographical distances and enabling teams, whether local or remote, to engage in discussions and decision-making with immediacy. What makes them indispensable is their capacity to capture non-verbal cues, encompassing facial expressions, body language, and tone of voice, enriching the depth of communication. Beyond mere words, video streaming opens the door to visual engagement, allowing for the seamless sharing of presentations and visual aids, which significantly enhance the quality of the discourse. This form of communication is vital in our interconnected world, where teams often span across various time zones, promoting effective global collaboration. Furthermore, audio and video streaming encourage active participation, as the visual connection fosters a sense of presence and engagement that minimizes distractions. In addition, they play a pivotal role in educational settings, offering interactive and immersive learning experiences. Whether building trust, recording meetings for future reference or elevating presentations, audio and video streaming are essential components in modern meetings, reshaping the dynamics of communication.

(2) **Interaction Type (Atomic Unit)**: Every meeting lacks a sense of fulfillment without the inclusion of interactive elements beyond just audio or video. We facilitate this by offering users specific Interactive Atomic Units shown in Figures 3 and 4, which can be better understood through an explanation of the various atomic units we have currently outlined in our software.
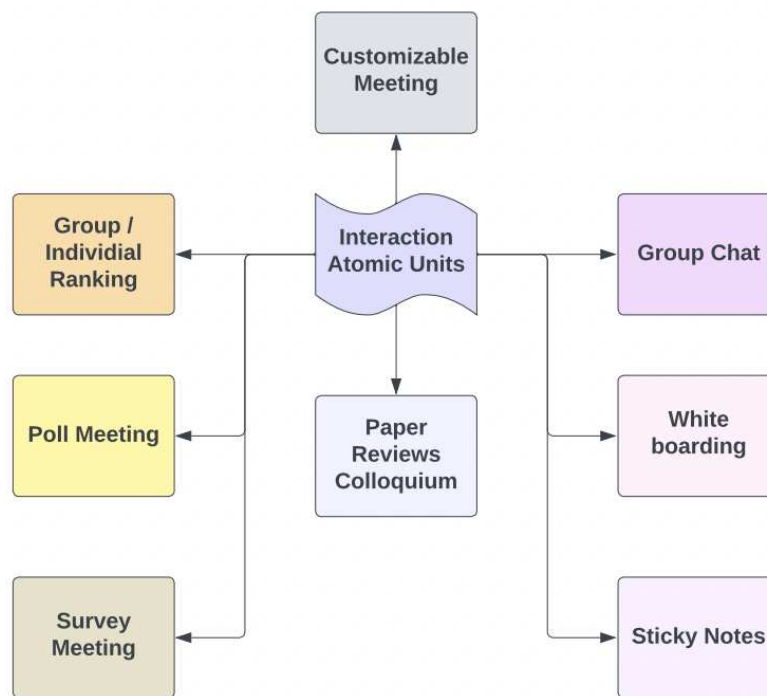


Fig. 3. Interaction Atomic Units

Each interaction unit is associated with its unique 'interactionObject' that encompasses all the attributes integral to its interaction definition. For instance, a basic group chat would include attributes such as 'isReply,' 'from,' 'visibleToAll,' 'text,' 'hasAudio,' 'audioEncryption,' 'hasVideo,' 'videoEncryption,' and 'sentiment' within its interactionObject. In a similar manner, a StickyNote interaction would possess its own distinct interactionObject, comprising attributes like 'postedBy,' 'postedText,' 'color,' 'x-coordinate,' and 'y-coordinate.' Notably, all these various interactions (e.g., WhiteBoarding, GroupChat, PollMeeting, Ranking, among others) also encompass a 'ResultObject.' This serves as the database logging format, containing significant parameters that can be queried for reference or used as logs to discern essential metrics. Additionally, each interaction has its 'SettingsObject,' housing parameters that may be necessary for configuring the meeting.

When a user adds a meeting to the event, they are prompted to provide details for all the parameters specified in the settings configuration. These details are then recorded as an entry in our backend system for future reference and utilization.

(3) **Data Persistence**: Data persistence in this context involves storing and retaining interaction data for long-term reference and analysis. It plays a key role in maintaining a historical record of user interactions, ensuring accessibility for reporting, analytics, and tracking user behavior. This data storage encompasses details of various interaction types, 'ResultObject' logging, user-configured meeting settings, and historical references, which are vital for troubleshooting, support, and decision-making. Data persistence enables the platform to harness historical data effectively, improving user experiences, and platform performance, and identifying trends and issues.



Fig. 4. Interaction Atomic Unit with Data Points

## 2.3 Event Monitoring - Defining Orders and workflows

The activities related to creating, updating, and deleting meetings are overseen by the event orchestrator integrated into our software. A sample event creation page looks like below. User has to provide details on all the attributes mentioned. During the process of creating a meeting, users have the option to reference data pointers from the resultObjects/interactionObjects of other meetings within the same event. This can be visualized as an example in 5, This feature offers greater flexibility and facilitates interconnections between meetings within the event. Consequently, the event orchestrator ensures that the meetings are arranged in a specific order, effectively establishing relationships between different meetings in the event. In simpler terms, you can envision different meetings within an event as distinct entities in a relational model, where a primary attribute of one meeting is linked to a foreign key from the result or configuration of another meeting. Figure 6 adds more meaning to this.



Fig. 5. Meeting Creation

The event orchestrator, a pivotal component of our software, serves as the central authority for managing and coordinating various activities within the event. It acts as the conductor, ensuring that meeting creation, updates, and deletions occur smoothly and coherently. Additionally, the event orchestrator plays a critical role in maintaining the integrity and order of meetings within the event, facilitating seamless data sharing and interconnectedness. By overseeing these processes, it enhances the overall organization, efficiency, and coherence of events, resulting in a more robust and streamlined user experience.
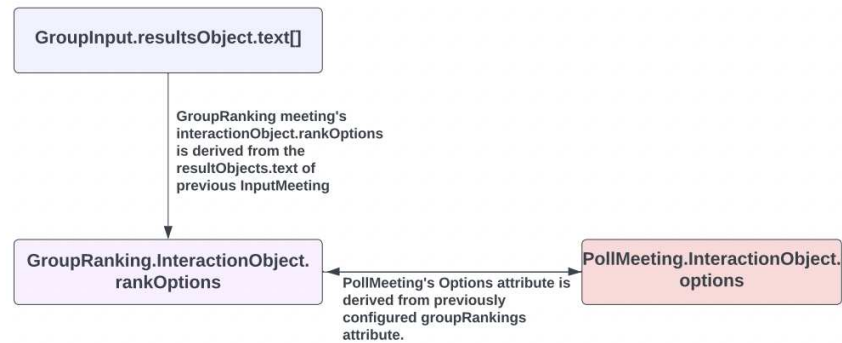
Fig. 6. Data Flow between Meetings in Event

## 2.4 Additional broad-level interaction pointers

The collection and analysis of user interaction metrics are fundamental to improving the user experience on a digital platform. It could be summarized as below:

(1) **Metric Collection**: The first step is to gather specific data points that shed light on how users are engaging with the platform. This can include tracking actions as straightforward as how often users click on certain features, like the "update an event" button. Such information is crucial for understanding which parts of the platform are most frequently used or encountered by users.

(2) **User Support Interaction**: Another valuable aspect of metric collection involves tracking user interactions with support bots or help features. This data helps identify pain points in the user experience where users may be struggling to navigate or understand certain features. For instance, users reaching out to support bots could indicate a need for improvements in user guidance or feature explanations.

(3) **Active Period Logging**: Examining the active period log provides insights into how long users remain engaged with the platform and whether they are following the expected user journey. This data can reveal if users are successfully completing tasks or if they are getting stuck at certain points in their interaction.

(4) **Machine Learning Analysis**: With a wealth of collected metrics, machine learning models come into play. These models process and analyze the data to identify patterns, anomalies, and opportunities for user assistance. Machine learning can reveal trends in user behavior and provide insights into when and where users might be encountering difficulties.

(5) **User Assistance and Enhancement**: Once the machine learning models have identified key points in the user journey where users commonly face challenges, the platform can respond proactively. This might involve suggesting relevant tips or hints to users when they seem stuck, guiding them through complex processes, or offering recommendations to enhance their experience. This approach aims to make interactions more seamless and user-friendly.

In essence, this data-driven approach to understanding user interactions and deploying machine learning models allows for a more personalized and supportive user experience. By identifying pain points and providing timely assistance, the platform can enhance user satisfaction and engagement, ultimately leading to a more successful and user-friendly digital environment. How this data could be used to boost the performance of the system

## 2.5 Some Special Pointers

*2.5.1 Breakout Rooms.* One of the other important functionalities of our application is facilitating breakout rooms. Breakout rooms are virtual spaces created within a larger online meeting or classroom, where smaller groups of participants can collaborate and interact with each other. This allows for more focused discussions and group activities, and can improve participation and engagement among participants. Some advantages of breakout rooms include:

(1) **Increased collaboration and engagement**: Breakout rooms encourage participants to interact and collaborate with each other in a smaller, more focused setting. This can lead to increased engagement and participation, and can also foster a sense of community among participants.

(2) **Improved learning outcomes**: Breakout rooms can be used to facilitate group activities and discussions, which can help to reinforce learning objectives and improve retention of information.

(3) **Enhanced problem-solving skills**: Collaborating in a small group setting can encourage participants to think critically and develop problem-solving skills.

(4) **Increased opportunities for feedback**: In breakout rooms, participants can receive feedback and guidance from their peers and instructors, which can help them to improve their skills and understanding of the material.

Overall, breakout rooms can be a valuable tool for online meetings and classrooms, helping to enhance collaboration, engagement, and learning outcomes.

*2.5.2 Project Colloquium.* Colloquium is a web interface that allows users to publish their in-progress work, encompassing everything from scholarly papers to presentations. and leave reviews of their peers' submissions. Moreover, it enables the reviews themselves to be reviewed in the form of a Reddit-inspired (recursive comment replies to comments, with the ability to upvote or downvote any given comment) comment system.

The origins of Project Colloquium trace back to an independent research initiative, embodying a pioneering vision in scholarly communication. The Colloquium platform introduces an inventive model of peer review, where submitted works are subject to evaluation through peer reviews that are themselves open to scrutiny through peer reviews. This recursive framework brings an essential layer of accountability to the review process, while still retaining the indispensable human element at its core.

Diplomat [9] a sophisticated bot wielding sentiment analysis, textual matching, and natural language processing. This arsenal of tools empowers Diplomat to deliver automated assessments on the value of peer reviews. The seamless integration of Diplomat within Colloquium amplifies the feedback loop, providing reviewers with more refined and insightful analyses, ultimately enhancing the overall quality of the peer review process.

## 2.6 Frontend Architecture

The frontend of our application is developed using ReactJS. There were some important considerations about choosing ReactJS. It is a popular and widely used JavaScript library for building user interfaces. It has a large and active community, and thus there are many resources available for learning and troubleshooting.It provides a component-based architecture that allows for modular and reusable code. This makes it easier to maintain and scale the application over time. It also is designed to work well with other JavaScript libraries and frameworks, which makes it a good choice for integrating with other tools and services. ReactJS provides a virtual DOM that updates only the necessary parts of the user interface, which can improve the performance and speed of the application. It encompasses the use of websockets, allowing for dynamic and real-time communication between the client and the server.

*2.6.1  Websockets.* WebSockets [7] provide a persistent, full-duplex connection between a client and a server, allowing for real-time data transfer. We primarily use SocketIO [13] since it supports Multiplexing allowinf us to use different channels under the same socket connection. Socket.IO is a JavaScript library that enables real-time, bidirectional and event-based communication between the browser and the server. Our backend uses Flask, and thus we employ the Flask-SocketIO library to easily implement WebSockets in Decidio 2.0. Our React frontend uses the "socket.io-client" library to establish a Socket.IO connection with the server.

Both WebSockets and Socket.IO are useful for building real-time applications that require frequent communication between the client and the server. While WebSockets provide a low-level protocol for bidirectional communication, Socket.IO adds a layer of abstraction on top of that, making it easier to work with and providing additional features such as automatic reconnection, message buffering, and more.

*2.6.2  WebRTC.* One of the very important functionalities of our application is the ability to stream audio/video simultaneously using the application to interact within different meetings in accordance with their specificity. We leverage WebRTC to enable our video/audio streams. WebRTC [1] (Web Real-Time Communication) is an open-source project that enables peer-to-peer communication between browsers and mobile applications using simple APIs. It allows real-time audio, video, and data communication over the internet without requiring any plugins or additional software.

WebRTC is built into modern web browsers, including Chrome, Firefox, and Safari, and can also be used in native mobile applications. It uses a signaling server to establish and manage peer-to-peer connections, which enables two or more devices to communicate directly with each other.

WebRTC is widely used for video conferencing, voice calling, screen sharing, and other real-time communication applications. It provides high-quality, low-latency audio and video transmission, making it ideal for real-time applications such as online gaming, live streaming, and remote collaboration.

WebRTC also provides a high degree of security, as it uses encryption to protect the data being transmitted. It is a powerful tool for building real-time applications that require high-quality audio and video transmission, low latency, and secure communication.

## 2.7   Backend Architecture

Our backend is built using Flask due to its lightweight and modular architecture, making it easy to develop and scale. Flask also offers a robust set of libraries and extensions that simplify development.
We have chosen MongoDB [2] as our database system for Decidio 2.0 because it offers a high level of flexibility, which is essential for a platform that supports various types of interactive meetings. MongoDB's document-based model allows us to store and retrieve complex data structures with ease. Additionally, its ability to scale horizontally makes it a great choice for a rapidly growing platform like Decidio 2.0.

One of the future works for Decidio is to introduce more types of meetings that users can create and customize using a simple drag-and-drop interface. To achieve this, we plan to use YAML [3] files to define and configure these meetings, and MongoDB's flexibility will enable us to store and manipulate these YAML files efficiently.

*2.7.1  Data Models and collections.* To use our application, users need a valid login. After logging in, they can create events that consist of multiple meetings in any order. We have defined different Meeting types for different domains, as mentioned in section 1.4. We want to make it easy for the host/admin to create new meetings by

dragging components they want, linking them, assigning the expected results and meetings data collection structure, with a workflow planner, and giving it a new purpose. We believe that creating a meeting interface that can produce yaml files based on the drag-and-drop and link actions and parse the yaml file to create the intended meeting can be a highly usable, cost-effective, and flexible solution.

Regarding data models and collections, we currently have three important collections: users, events, and responses. The users collection contains Personally Identifiable Information (PII) such as emailId and login mode, which can be Gmail, Facebook, or email sign-in. The Events collection will hold information about the events and the various meetings they contain. It tracks the status of the event and the meetings at any given time. Meetings that don't require a lot of storage space for storing responses or results, such as poll meetings or group rankings, will have their results stored within the Events collection. For all other meeting results and responses, they will be stored in the Responses collection.
In the near future, we plan to introduce collections that will store YAML metadata, which will allow for dynamic rendering of meetings.

## 3  INSTRUMENTATION AND METRICS

Instrumentation and metrics are important in our Decidio application for several reasons:

(1) **Monitoring application performance**: By instrumenting the application with metrics, we can monitor its performance in real-time and identify any issues that may arise. This enables us to proactively take action before the issue escalates, ensuring the application is running smoothly and efficiently.
(2) **Identifying and resolving bottlenecks**: Metrics help us identify bottlenecks and areas of the application that may need optimization. By analyzing metrics, we can determine which areas of the application are experiencing high load and optimize them to improve overall performance.
(3) **Measuring user experience**: Metrics can be used to measure the user experience and identify any issues that users may be facing. For example, we can track the time it takes for pages to load or how frequently users encounter errors. By identifying and resolving these issues, we can improve the user experience and ultimately increase user satisfaction.
(4) **Data-driven decision making**: By collecting and analyzing metrics, we can make data-driven decisions regarding the development and operation of the application. This allows us to make informed decisions about where to allocate resources, which features to prioritize, and how to optimize the application for performance and user experience. We could gradually work towards introducing bots that could help the users create meetings with very little manual intervention.

Overall, instrumentation and metrics provide valuable insights into the performance and usage of the Decidio application, enabling us to optimize its performance, improve the user experience, and make data-driven decisions.

In the near future, we intend to implement an orchestrator that will facilitate a workflow-based meeting process, resulting in a more organized and structured approach. With this orchestrator, we will have control over the movement of data throughout the meeting workflows through secure channels, enabling us to create a more systematic system while still maintaining maximum flexibility. There are unanswered questions on what do we collect in order to enable experimentation?, How are these preserved and used?, How do these data become connected with survey instruments? that we are trying to investigate.

Fig. 7. Decidio 2.0 Architecture

## 4 FUTURE ENHANCEMENTS

### 4.1 Transforming to a Micro-Service Architecture

Microservices architecture is an approach to software development that involves creating small, independent services that work together to form a larger application. In this architecture, each service is responsible for a specific task or functionality and communicates with other services using APIs (Application Programming

Interfaces). Adopting a microservices architecture can be highly beneficial for the Decidio application. Here are some of the ways in which this architecture can improve the application:

(1) **Scalability and High-availability**: With a microservice architecture, we can scale individual services independently based on demand. This is particularly useful for the Decidio application since it involves hosting events and meetings. With the ability to scale up the services required for a particular event, we can ensure that the application can handle increased traffic and workload during peak periods.

(2) **Flexibility**: Adopting a microservices architecture would provide Decidio with the flexibility to choose the best tools for each service. For instance, different services can be developed using different programming languages and data storage technologies, depending on their specific requirements. This would enable us to leverage the strengths of each technology and create a more robust and efficient application.

(3) **Fault isolation**: In a microservices architecture, each service operates independently of the others. This means that if one service fails, it does not bring down the entire application. The remaining services continue to function normally, ensuring that the application remains available to users. With the Decidio application, this would ensure that events and meetings can continue as scheduled, even if one or more services experience issues.

(4) **Improved development speed**: With a microservice architecture, development teams can work on different services independently of each other. This enables teams to work on different parts of the application simultaneously, reducing development time and improving overall productivity.

(5) **Easier maintenance**: With a microservice architecture, each service is independent of the others, making it easier to maintain and update the application. Changes to one service do not affect the others, making it easier to make changes without worrying about breaking the entire application.

In conclusion, adopting a microservices architecture for the Decidio application can provide significant benefits, including scalability, flexibility, fault isolation, improved development speed, and easier maintenance. By leveraging this architecture, we can create a more robust and efficient application that meets the needs of our users.

## 4.2   Generalized Frontend Workflow

The Decidio application is designed to be a platform for hosting events and meetings, and as such, it requires a generalized frontend workflow that can accommodate the different requirements of various events. This frontend workflow must be flexible enough to allow users to customize their event pages and create workflows that meet their specific needs.
To achieve this, we plan to develop a modular and customizable frontend architecture that enables users to create and manage their event pages easily. The workflow will provide different components that users can use to create their pages, such as forms, calendars, chatbots, video conferencing tools, and more. These components will be designed to be modular, allowing users to select and configure them based on their specific needs.

In addition, we are exploring the idea of making Decidio a PaaS (Platform as a Service) that can run ML (Machine Learning) engines (Plug and Play engines with minimal configuration changes needed) with the necessary hardware support. This would enable users to perform complex analyses and make data-driven decisions during their meetings and events. For example, users could use ML to analyze responses to polls or surveys in real-time and use this information to make decisions during their meetings. They could also use ML to identify trends and patterns in large datasets, which could help them make more informed decisions.

To make this possible, we will need to ensure that Decidio has the necessary infrastructure to support ML. This will require us to provide access to high-performance computing resources, such as GPUs (Graphics Processing

Units) and TPUs (Tensor Processing Units), as well as the software tools required for ML development and deployment. We will also need to ensure that Decidio is secure and compliant with industry standards and regulations, such as GDPR (General Data Protection Regulation) and HIPAA (Health Insurance Portability and Accountability Act).

In short, the Generalized Frontend Workflow and the PaaS architecture with ML support are important features that will enable Decidio to be more flexible, customizable, and powerful. By providing users with the tools they need to create and manage their events, as well as access to high-performance computing resources, we can help them make more informed decisions and achieve their goals more efficiently.

## 4.3 Smart Caching

We will likely implement an intelligent caching layer, such as Redis[5], to store metadata variables and improve data retrieval speed compared to directly fetching data from the database. This would require setting up a periodic data synchronization process, which could be automated through a cron job and a manual in-application trigger. This approach would further enhance the availability and reduce latency of the application.

## 5 CONCLUSION

We are concurrently shepherding several small-scale pilot studies of participant interaction practices, each intended to illustrate one or another of the functionalities of the Decidio 2.0 design. Not least among these is the deployment of Decidio as a front end of structured review meetings, with content managed in our Colloquium system and in support of studies on the curation of reputation and rankings. All of these activities help us evaluate the performance of Decidio 2.0 as well.

## 6 ACKNOWLEDGEMENTS

## REFERENCES

[1] Harald Alvestrand, Michael Bannister, Eunsoo Choi, Boris Cubrilo, Björn Einarsson, Oscar Hagström, Magnus Holmer, Christian Huitema, Adrian Johansson, Jonas Jonsson, Peter Josefsson, Ari Keranen, Oleg Lyutkevich, Alexey Melnikov, Andreas Moen, Biswanath Nandy, Julian Pettit, Eric Rescorla, Philippe Saint-Andre, Henning Schulzrinne, Jürgen Singer, Michael Sokolov, Pasi Srisuresh, and Jim Weinhold. 2021. Web Real-Time Communication (WebRTC): Protocols and APIs.  https://datatracker.ietf.org/doc/rfc8825/

[2] Kyle Banker, Douglas Garrett, Peter Bakkum, and Shaun Verch. 2016. *MongoDB in action: covers MongoDB version 3.0.* Simon and Schuster.

[3] Oren Ben-Kiki, Clark Evans, and Brian Ingerson. 2009. Yaml ain't markup language (yaml™) version 1.1. *Working Draft 2008* 5 (2009), 11.

[4] Michele Biasutti. 2017. A comparative analysis of forums and wikis as tools for online collaborative learning. *Computers & Education* 111 (2017), 158–171.  https://doi.org/10.1016/j.compedu.2017.04.006

[5] Josiah L. Carlson. 2013. *Redis in Action.* Manning Publications Co., USA.

[6] João Carneiro, Diogo Martinho, Goreti Marreiros, and Paulo Novais. 2019. Arguing with behavior influence: a model for web-based group decision support systems. *International Journal of Information Technology & Decision Making* 18, 02 (2019), 517–553.

[7] Ian Fette and Alexey Melnikov. 2011. The WebSocket Protocol.  https://www.rfc-editor.org/rfc/rfc6455.html

[8] George Gorry and Michael Scott-Morton. 1971. A framework for management information systems,. *Sloan Management Review* 13 (01 1971).

[9] Kevin Hogan, Annabelle Baer, and James Purtilo. 2021. Diplomat: A conversational agent framework for goal-oriented group discussion. In *International Conference on Group Decision and Negotiation.* Springer, 143–154.

[10] Kevin Hogan, Fei Shan, Monikka Ravichandran, Aadesh Bagmar, James Wang, Adam Sarsony, and James Purtilo. 2020. Decidio: A Pilot Implementation and User Study of a Novel Decision-Support System. In *Group Decision and Negotiation: A Multidisciplinary Perspective*, Danielle Costa Morais, Liping Fang, and Masahide Horita (Eds.). Springer International Publishing.

[11] Emily J. Hurst. 2020. Web Conferencing and Collaboration Tools and Trends. *Journal of Hospital Librarianship* 20, 3 (2020), 266–279. https://doi.org/10.1080/15323269.2020.1780079

[12] Juan Carlos Leyva López, Pavel Anselmo Álvarez Carrillo, Diego Alonso Gastélum Chavira, and Jesús Jaime Solano Noriega. 2017. A web-based group decision support system for multicriteria ranking problems. *Operational Research* 17 (2017), 499–534.

[13] Rohit Rai. 2013. *Socket.IO Real-Time Web Application Development.* Packt Publishing.

[14] Atiq W Siddiqui, Syed Arshad Raza, and Zeeshan Muhammad Tariq. 2018. A web-based group decision support system for academic term preparation. *Decision Support Systems* 114 (2018), 1–17.

[15] Jos van Hillegersberg and Sebastiaan Koenen. 2016. Adoption of web-based group decision support systems: experiences from the field and future developments. *International journal of information systems and project management* 4, 1 (2016), 49–64.