



UNIVERSITY OF  
**WATERLOO**



**Computer Science /  
Software Engineering  
Capstone Design Projects 2021**

May 13, 2021



## Teams

Virtual Experience	1	Tutturu
	2	Punctuation
	3	WaterlooWave
	4	GeoCache
	5	JKKody
Journaling	6	JAMN
	7	GoalGoat
	8	Bery
Autonomous Systems	9	AutoDash
	10	OASIS
Software Systems	11	Bfame
	12	SEXXI Goose
	13	Censorship Circumvention
Game Dev	14	Jellyfish
	15	Elephant
	16	unFYDP
Games	17	AIGame
	18	GoSudoku
	19	PlayMahjong
Document Editing	20	Burrows
Pronunciation	21	Pronounce
Shopping	22	Pronto

# Projects

Tutturu: shared browsing with friends

An improved online lecture note-taking experience

WaterlooWave App

A geocaching inspired social media app

Hallway Hello: An App to Meet New People Online

Introspect: a journalling app

A social journaling app for long-term goals

mygrocery: Reducing domestic food waste

Pipeline for Car Collision Aggregation and Labelling

Making Drone Development Easier

Diago: A Testing Framework for K8s Services

Rust Compiler Patches

Researching practical circumvention tools

Making multiplayer games easy

Mesa: Host game night online

Poseidon: 3D Humanoid Asset Pipeline

A card game that uses procedural card generation

GoSudoku: a real-time, collaborative Sudoku game

An online multiplayer Mahjong platform

Yada: Yet Another Docs App

Syllable: Say everyone's name (correctly)

Data-driven product comparison website

# 1 Tutturu: shared browsing with friends

## Team Tutturu

Social distancing is hard. The inability to see your loved ones in person is a heart-wrenching problem that has been faced by online friend groups, couples in long-distance relationships, and family members studying abroad. The COVID-19 pandemic has significantly increased the number of people that face this problem; something needs to be done.

Unfortunately, many online social activities require a lot of commitment:

competitive multiplayer games are not fun for everyone, teaching the rules of your favourite board game takes time, and an unexpected small turnout for trivia night can be frustrating - you are reliant on everyone showing up. We need more inclusive, simple, and flexible online activities.

We are building a shared browser platform that allows multiple people to control a shared web browser in a virtual machine. The desktop display is streamed back to all the participants, enabling friends to enjoy online content in synchrony. Anime shows, YouTube videos, TikTok memes: if it's on the internet, you can enjoy it together on Tutturu.

Shared browsing has two major advantages over alternatives like screen share. The first advantage is the ability to transfer control to friends, leading to a more interactive and seamless experience. The second advantage is that there is no "host"; nobody has to share their screen, which means higher streaming quality, more privacy, and nobody needs to stick around longer than they want to.

We launched our MVP, Tutturu on December 25th, 2019. At the time of writing, we have over 1.2 million user signups and over 2 million hours of cumulative session time. We have over 3 thousand subscribers that pay \$5 per month, who spend an average of 22 hours per month on our platform.



Ambareesh Balaji, Declan Goncalves,  
Philip Scott

[tutturu.tv](https://tutturu.tv)

## 2 An improved online lecture note-taking experience Team Punctuation



Nathan Ko, Yixuan Wang, Jaxon Lin

As COVID hit in 2020, most, if not all, university students moved from an in-person to an online learning experience. In our initial note-taking habit survey<sup>1</sup>, about 97% of our respondents took notes digitally. A majority (86%) of them took video-related notes. Of those students, most (71%) had more than one application open while doing so: at least one for note-taking and another for the lecture video. Reviewing

notes was another hassle, as students had to explicitly jot down a timestamp for future reference and manually scrub to that point in the video when re-watching lectures.

As such, we built Punctuation, a web application that focuses on providing students an improved online learning experience. Punctuation eliminates the need to open several applications while watching online lectures by combining a medium player with a side-by-side editor. Moreover, a user can effortlessly control medium playback with Punctuation's integrated keyboard shortcuts, while typing their notes, without them having to remove their hands from the keyboard. They can also mark a note passage with the time in the video that relates to the passage. This timestamp later allows a student to simply, on click, jump to a relevant section in the medium. Despite university students taking remote classes being the application's primary target, Punctuation remains useful for anyone who wants to take notes while watching videos online.

A major challenge of this project lies in promoting its simplicity of use within a saturated marketplace of note-taking applications. As most university students may already have their go-to editor, Punctuation attempts to entice students with its timestamping feature and integration between lecture videos and notes. These key points are functionality we have not yet seen in existing note-taking applications. Punctuation also has an affiliated Chrome extension that allows users to source a video from any website, with the press of a button, on which to take notes. This allows for broader video support, even when video links are not directly available.

<https://punctuation.space/>

---

<sup>1</sup>conducted in June 2020, with a sample size  $N = 46$  of mainly university students

### 3 WaterlooWave App Team WaterlooWave

As we experience a worldwide shift where social distancing is becoming the norm, the ways people are able to safely connect with one another have become limited. Loneliness during the COVID-19 pandemic has become a major concern as more and more people are hiding in their homes in isolation. With physical distancing and the disruption of academic terms, students are facing a number of challenges including adjusting to new routines and seeking outlets to boost physical and mental wellness.

WaterlooWave is an app that aims to help alleviate the loneliness some students might face during quarantine. The goal is to create a sense of connectedness, during a time where social interaction is limited, as it plays an essential role to the emotional well-being and academic success of students.

Our approach is to create a mobile platform that allows students to connect with other students residing in the same region for small activities. Students will have the ability to create their own activities as well as join activities created by other students within their city. Types of activities can range from virtual activities such as gaming, to outdoor, in-person (and legal) ones such as tennis, photography, biking, etc. . . Upon joining an activity, the student is placed in a group chat where they can further discuss details with other group members.

The main advantage of this design over major alternatives such as social media sites like Instagram and Facebook is that it allows Waterloo students to connect more easily, in an organized way, based on location, year, faculty, and/or program. The app also encourages students to interact in a safer and more personal manner as the activities are implemented in small groups.

We will evaluate the success of our application based on metrics such as the number of activities created, the number of participants and active users, and retention—how many users are returning to the app.



Sarah MacLean, Jerome Alve, John Weber

## 4 A geocaching inspired social media app

### Team GeoCache



Krisztian Palvolgyi, Milan DiGiuseppe,  
Jason Xu, Will Brown

Our objective with Geo is to encourage people to get outside and interact in a fun and safe way. Fresh air and socialization are important and often neglected determinants of mental health, especially in the context of the pandemic.

Geo is a mobile app where users create and discover “caches” as they explore their neighbourhoods. Caches are like posts in traditional social media, and users are encouraged to be creative with what they post! To promote outdoor activity, users need to walk up to a cache before they can interact with it. With Geo, users are not only socially distancing, but also temporally distancing!

Caches will stick around awhile where they have been created, so users can interact with them at different times. This kind of socialization requires getting outside, but doesn't need people to be face-to-face.

Geo has a unique position in the app marketplace when considering the dimensions of socialization and outdoors use. Traditional social media apps like Instagram or Twitter are great for communicating with friends and family, but are often used at home on your couch or at your computer during the pandemic. An app like YikYak, a social media where users contribute to a community feed within a 5 mile radius, is getting closer to our idea. On the other hand, Pokemon Go is great at promoting users to get off their couch and walk around, but socialization is limited and focused on the Pokemon fanbase. Geo aspires to be a fun way of socializing as well as spending time outdoors!

To measure Geo's success, we will conduct a user study on a Minimum Viable Product that has enough features to provide value to the user. The user study will feature a cognitive walkthrough to evaluate the user experience of Geo as well as open-ended discussions and feedback about current and proposed features of the app. The feedback will be analyzed and triaged to implement an app that is polished enough for a future App Store / Google Play release.

[geocacheapp@googlegroups.com](mailto:geocacheapp@googlegroups.com)

## 5 Hallway Hello: An App to Meet New People Online Team JKKody

As of Fall 2020, the University of Waterloo has 42,000 students enrolled. Because of the pandemic and university transitioning to virtual class, the feeling of isolation for all 42,000 students grows. By having classes online, students miss out on one of the best aspects of a university experience: meeting new people and making new friends. Hallway Hello replicates casual hallway conversations between students virtually. Our application aims to stimulate the social aspects of a university degree by allowing users to spontaneously meet others and make meaningful connections with classmates. By grouping students together, they can meet online with fun icebreakers inside the chat rooms to create engaging conversations.



Judy Liu, Calvin Thye, Kody Tang

Hallway Hello is a Microsoft Teams application that can be added as a tab to any existing Team in the University of Waterloo. We chose Microsoft Teams because this is the official communication platform of the University and to leverage the existing chat and video calling features. The key design challenges will be designing matching algorithms, and integrating our application with Azure. This is the first time our entire team has developed an application for Microsoft Teams and in C#.

The main advantage is this application is specifically targeted at University of Waterloo students. A key issue with meeting people online is trust and safety. For example, Omegle.com is a popular application to meet strangers online but is commonly known for the harmful content perpetuated. We will control trust issues by limiting the circle of users to your classmates.

The main criteria in this project category is a function of the number of users and the amount of engagement each user has with the software. We will be looking at a number of metrics in order to measure this such as: number of monthly users on the platform and average minutes spent video calling with matches. Ultimately, we are hoping for thousands of light users who can leverage this platform to meet new students and connect later on with various social media outlets.



## 6 Introspect: a journaling app

### Team JAMN



Jiayi Zhou, Andrew Lamb,  
Michael Leng, Nicholas Good

The reflective process of writing down one's thoughts and memories has been proven to have many benefits. Journaling is an effective way in which one can express their innermost thoughts and reflect on the events in their lives, both life-changing and mundane. It is by recording and reflecting upon these snapshots of their lives that one can truly gain insights into their personal growth.

Introspect is a journaling app that aims to encourage daily reflection

through daily prompts, quick responses, and a suite of retrospective features to help users look back.

Its aim is to lower the barrier of entry for beginners who want to develop this habit, as well as provide interesting insights and an enjoyable experience to those who want to enhance their journaling experience.

Introspect's core functionality centers around prompts that act as inspiration for the users' thoughts and give direction to their entries. Using a categorization based on these prompts, *Statistics* provide data about the user's writing habits and *Memories* surface previous entries on the home page that may be relevant. *Statistics* and *Memories* encourage the user to reflect on their past writing.

In addition to these reflection tools, Introspect has a rich feature set for text editing, including the ability to add handwriting to each entry.

Introspect is available on both iOS and web to allow users to write wherever and whenever they are most comfortable.

With the combination of a curated experience and powerful text editor, Introspect is in a unique position within the digital journaling space. Our goal is for Introspect to become an appealing option for both new and experienced journaling enthusiasts. Some good metrics to evaluate this objective are: total number of users, user retention, and overall sentiment towards the user experience.

<https://www.introspectjournal.com>

## 7 A social journaling app for long-term goals Team GoalGoat

Many people have long term, “pipe dream”-esque goals that they wish to pursue but have trouble achieving due to fear of failure or lack of motivation. Today, many people use mobile applications to handle goal planning. Existing long term goal planning apps such as “Dreamfora” or “Life Goals” have over 100,000 downloads each, showing the demand for apps in this space. However, existing goal tracking apps on the market do not have a social network component.

Some theories show that key criteria of successful goal achievement are hard to fulfill without a social community. By providing a platform for users to post about their goal and

interact with others, GoalGoat creates a community that engages the user and provides them with a sense of accomplishment. Interactions on GoalGoat include posting stickers and notes on other users’ journal entries to create a supportive environment. We also use the concept of anonymity and pseudonymity to allow users to express their authentic thoughts without being judged by pre-existing biases. These features combined allow users to share their goals in a community of people who motivate each other to achieve greater things.

We implemented GoalGoat with React Native, as this platform is well documented, is widely supported, has many extensions, and allows cross-platform development with native performance. We chose MongoDB to store our data, since this kind of database provides better flexibility. Our decision to make GoalGoat exclusively a mobile platform allows users to readily access and modify goals as they change. In addition, being mobile motivates users to be more responsive to interactions on the platform, fostering a more active and engaged social community.

To gauge the success of GoalGoat, we will gather the number of installs, journal entries, stickers and notes given, and times any baby step has been completed. These statistics represent the size of the active social community and level of engagement with other users.



Bo Yuan Tan, Doris Zhang, Rachel Lang, Mengfei Liu

## 8 mygrocery: Reducing domestic food waste

### Team Bery



Beverly Vaz, Emily Lin, Ruiying Bu,  
Yingjie Guo

The National Zero Waste Council estimates that Canadians waste 2.2 million tonnes of food annually, which corresponds to 17 billion dollars worth of food being wasted each year. However, they also believe that over half of this wastage can be avoided. The issue with food waste is that it wastes not only the items

themselves, but also all of the resources used to produce, grow, and distribute them.

This information inspired us to create a mobile application that helps as many people as possible improve their habits and behaviour in order to reduce their food waste.

Our application attempts to achieve these goals by providing a way for users to better manage their grocery planning, current stock of food, and to track their wastage. An example feature is tracking items that a user has wasted and, if they try to add that item to a new grocery list, the application will detect the past wastage, notify the user of this, and then prompt them to come up with a strategy to avoid wasting it again. The challenges for this project include designing an intuitive and user-friendly UI and planning out and conducting the entirety of our study virtually.

The advantage of our design is that our application uses principles from psychology such as Fogg's Behaviour Model, self-reflection, and goal-setting, to encourage users to improve their behaviour, which isn't present in any other application on the market. Our application is for all users who have an Android device and are interested in reducing food waste.

In order to measure the success of our application, we are conducting a user study. Throughout the study, we will actively consult with our users to understand the effect of our application on their habits around food waste management. We will also ask them for feedback on the UX of our application and make improvements accordingly.

[teambery21@gmail.com](mailto:teambery21@gmail.com)

## 9 Pipeline for Car Collision Aggregation and Labelling Team AutoDash

The autonomous vehicle market is expected to grow to 4 billion units by 2030. However, for this to happen, many problems related to autonomous vehicles must be solved. Automated collision avoidance is one such problem that requires large collision-oriented datasets. However, using high-end equipment, such as Google's \$7,500 per unit LIDAR, quickly grows prohibitively expensive for data collection.

The main objective of our project is to produce software that aims to advance research in automobile collision analysis and prevention.

Additionally, we wish to produce a high-quality dataset of annotated dashcam footage from a labelling tool that researchers can rely upon when designing and training their own accident prediction models.

We are doing so by creating a dashcam-video labelling tool that aggregates videos from online sites, such as YouTube, Reddit, and Imgur. The tool automatically uploads the labelled video metadata to a public database, where it can easily be fetched by researchers and used for a variety of applications.

The advantage of dashcams is they are both affordable and accessible to any driver. Therefore, they have recorded a wide variety of collisions and driving scenarios not limited to any specific geographical location or road conditions.

Simulations can provide more data synthetically, but great care needs to be taken to ensure the simulation represents reality. The result could easily lack many unique situations outside the developer's perspective.

We expect to aggregate a dataset of at least 200 dashcam video segments each 5s in length. Each segment will contain bounding-boxes and labels for all vehicles present as well as data about the collision itself.

With our dataset we expect to train an existing model architecture and achieve performance comparable with the state-of-the-art. Existing work has achieved 73.53% accuracy at 80% recall and 56.14% precision, while anticipating collisions on average 1.8559 seconds before the collision event on a video dataset created exclusively in Taiwan. We hope to achieve similar performance with data representative of global road conditions.



Chengzhong Meng, Curtis Phan,  
Chaitanya Varier, Jack Williamson,  
Nick Geofroy, Ben Upenieks,  
Vincent-Olivier Roch

[autodash.github.io](https://autodash.github.io)

# 10 Making Drone Development Easier

## Team OASIS



Bohan Shi, Xiao Zhou, Daniel Xu,  
Frank Kang, Clair Hu

Drones are used in an unprecedented number of ways since the advent, with millions of people using them for agriculture, journalism, photography, courier services, etc. However, drone application development itself includes lots of hardware interfaces and inconsistent format aligning, which makes it time-consuming for the developers, especially those who just begin drone development, to successfully finish an application. Seeing this problem, we propose to develop some solutions that make the development of drone applications easier and make the drones safer at a lower altitude.

The goal of this project is to build an SDK that simplifies the development of automated drone and drone-related applications. It can also smooth the learning curve for new drone developers and prompt code base sharing.

OASIS, short for Obstacle Avoidance with Spontaneous Itinerary Strategies, comes from one of our obstacle avoidance features, which allows multiple strategies (algorithms) to run simultaneously. For example, both LIDAR and regular cameras with AI can be used to determine collision probabilities and/or paths independently, which are then synthesized for a final decision.

We support a set of features including: A set of unified interfaces for sensors and Mavlink based flight controllers; integrated obstacle avoidance features; game engine based SITL visualizer and simulator; a new simplified console command interface; an internal low-level control API; application manager; a Web-based UI API; full ROS Support and open source.

We have four target user groups: (1) To new drone developers: reduce the learning time. (2) To senior drone developers: simplify development and reduce the maintenance cost. (3) To researchers and app developers (use drones only as vehicles): don't need to worry about unrelated low-level stuff and compatibility. (4) To advanced operators: help to add in new features including obstacle avoidance on their vehicles.

<https://gitlab.tuotuoqzs.com/droneoa>

## 11 Diago: A Testing Framework for K8s Services Team Bfame

Over the recent years, Kubernetes has emerged as one of the industry standards for managing services at scale. One of the major advantages it brings is the large ecosystem of open source tools that improve productivity. Despite this wide selection, we found that there is a distinct lack of tools that target a very important aspect of services on Kubernetes: load testing.

Today, engineers perform load testing on Kubernetes services by juggling a variety of tools that individually handle running tests from outside the cluster, monitor application performance, and simulate random failure. By consolidating the functionality of these tools and providing a simplified interface, we believe that we can create a single tool that drastically improves load testing productivity.

Diago is an open source load testing framework designed for Kubernetes that combines all the aforementioned features, and more, into a single application. It is meant to generate load for services both inside and outside a Kubernetes cluster, while monitoring service performance. It can also simulate random failure for services inside Kubernetes during a load test, which helps determine various key metrics like total time to recovery. In addition, one of the main benefits of using Diago over alternative frameworks is that it is designed to efficiently use a cluster's resources by being elastic and leveraging Kubernetes's API, which means that operation costs are kept low. Knowledge of scheduling and distributed systems is applied in this respect to keep track of resource demands, and coordinate between nodes. Finally, Diago is designed to be easy to set up and simple to use, allowing engineers to integrate it effortlessly and become immediately productive.

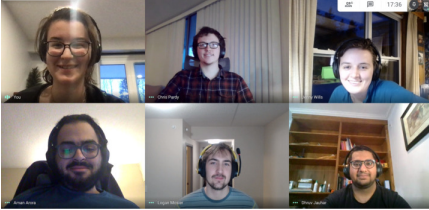
In order to determine whether Diago achieves its goals, we plan to analyze productivity by comparing the setup time of Diago versus that of a combination of other individual tools which combine to provide an intersecting feature set. Ultimately, our goal is to improve an engineer's insight into their services by load testing them before they go to production, thereby reducing outages.



Bill Qi, Frank Liu, Armanit Garg,  
Michael Cai, Ethan Xue

<https://github.com/t-bfame/diago>

## 12 Rust Compiler Patches Team SEXXI Goose



Roxane Fruytier, Chris Pardy,  
Jenny Wills, Aman Arora,  
Logan Mosier, Dhruv Jauhar

Rust is blazingly fast and memory-efficient: with no runtime or garbage collector, it can power performance-critical services, run on embedded devices, and easily integrate with other languages. And it is truly unique due to its system of ownership and borrowing. However, the current system is a little over zealous when dealing with structs or tuples whose individual fields are captured in a closure: the whole struct or tuple will be bor-

rowed by the closure if even a single one of its fields is captured. This capturing is more aggressive than is necessary and means that some safe (and otherwise valid) programs are rejected.

With precise path capturing implemented for closures, it will expand the set of valid programs recognized by the compiler, and the borrowing principles inside a closure will be more consistent with the borrowing principles outside closures.

The design approach taken was to improve the existing capture analysis for closures, allowing the compiler to have access to more information about the captures so that it can make appropriate capture decisions.

The RFC as it was written and approved had left certain questions to be answered during the implementation phase of the project. Some of these questions and topics that were not discussed in great detail possessed challenges both from an implementation and a language design perspective.

The main advantage of this solution/RFC over the existing closure capture analysis implementation, is that it unifies the rules of borrowing and ownership in and outside of closures. This extension to the language also allows more code that is safe (from a memory perspective) to be accepted by the language.

This feature is currently available on Rust Nightly release channel, with the goal to have it released in the Rust 2021 release. It can be enabled using `#![feature(capture_disjoint_fields)]`.

[https://github.com/rust-lang/rfcs/blob/master/text/  
2229-capture-disjoint-fields.md](https://github.com/rust-lang/rfcs/blob/master/text/2229-capture-disjoint-fields.md)

## 13 Researching practical circumvention tools

### Team Censorship Circumvention

Censorship circumvention, a practice that enables equal access to information under a censoring entity, is an academic field with many technical challenges. In a survey of the field, Tschantz et al. [1] identified a gap between theoretical and practical tools. We use dnstt to study the challenges of practical censorship circumvention.

dnstt, an open-source project created by David Fifield, uses DNS-over-HTTPS (DoH) to perform DNS tunneling for censorship circumvention. DoH shows promise as a medium for circumvention as its general adoption increases. However, before any implementation of DoH tunneling can be used against a real censor, the traffic it generates must be indistinguishable from normal DoH.

Our experimental analysis confirms that DoH tunneling using dnstt is in fact distinguishable from normal DoH. We wrote four attacks that accurately detect dnstt sessions, with the Maximum Packet Length Attack analyzing as few as 82 packets on average before detection. We further show that modifying dnstt to be censorship-ready reduces its usability. Notably, throughput is decreased by at least 27x and page load time is increased by at least 23x.

This project was advised by Cecylia Bocovich, a visiting researcher in Computer Science at the University of Waterloo, and a developer at Tor Project.

[1] M. Tschantz et al. 2016. SoK: Towards Grounding Censorship Circumvention in Empiricism. In IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016 (pp. 914–933). IEEE Computer Society

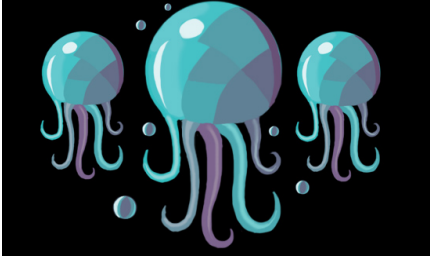


Carmen Kwan, Cathy Wang, Paul Janiszewski, Shela Qiu



## 14 Making multiplayer games easy

### Team Jellyfish



Guanzhong Chen, Michael Yaworski,  
Nicholas Throm, Doug Sawadsky, Run  
Tong Li

Making multiplayer games is hard, but it shouldn't have to be.

Almost every popular game these days supports multiplayer. Many people want to make multiplayer games — tutorials online on how to do so have hundreds of thousands of views. Despite this, game engines leave a gap in the market when it comes to multiplayer. They provide only basic tools to implement multiplayer, so it takes significant time and expertise. This is an obstacle to many game developers, especially be-

ginners and independent developers.

Jellyfish, a JavaScript game engine, aims to fix this problem. Jellyfish replaces the usual low-level tools for dealing with multiplayer, with intuitive high-level tools. The developer writes game logic like a single player game, and Jellyfish handles the rest. Jellyfish's automatic multiplayer synchronizes the game on a server and multiple clients, with almost no extra work from the developer.

Jellyfish does this by providing a game engine, and automatic multiplayer. The engine ensures the state of the game is always easily serializable, so the developer can focus on the game, instead of serialization. Then, automatic multiplayer serializes the games state, and synchronizes it between the server and the clients. The only thing the developer needs to do is enable multiplayer.

Nothing similar to this functionality is available in most other game engines. Most game engines only provide an API for handling networking. The few that have a high-level concept of multiplayer usually have it as an intermittently maintained extra feature that only makes things slightly easier. The only products similar to Jellyfish are expensive, closed-source, unofficial add-ons, while Jellyfish is a free open-source game engine designed specifically for multiplayer.

We plan to evaluate Jellyfish by asking developers how long they would expect to spend developing a very simple multiplayer game, then asking them to time how long it takes for them to develop one with Jellyfish. We expect it will take less time than predicted, a rare accomplishment in development. We will also gather qualitative feedback on the usability of Jellyfish.

<https://jellyfishjs.org/>

## 15 Mesa: Host game night online Team Elephant

In 2020, many countries have exercised strong lockdown and social distancing policies. This has made it hard to meet with groups of friends, and places where you would previously meet (cafes, bars, etc.) are not viable. This grants us the opportunity to allow friends to share a space digitally. Shared space allows groups of friends to enjoy things they would normally do together. This approach also grants us a lot of flexibility that

people would not have even if they met in person. They can drag and drop various game elements, create their novel games, or simply play an existing game. The lack of strict enforcement of rules allows players to be creative and modify games to their liking. It simulates a real-life environment of a table where groups might play cards or board games within a web browser.

Individuals can create game rooms through the website, and share direct links to friends so that they can join the room. Once within the room, users have full control of the table. The primary method of interaction is through widgets. Widgets are anything that can be placed on the table and manipulated. For example, some basic widgets that are included include tokens from popular games such as Chess and Checkers as well as more advanced widgets, such as a deck of cards. These items can be moved around the board as well as manipulated depending on the type of widget they are (eg. you can shuffle a deck of cards and deal cards to players). This combined with the ability to add game-specific widgets (such as a chessboard) allows users to play a multitude of games together in this virtual environment.

The main advantage of our platform is the frictionless experience, creating and joining a room with friends is effortless, and the rich customizability to create and enjoy games. Games can also be saved in a serialized format for use later. The user can additionally customize further by uploading their images to use as various game pieces. This allows for situations where users can design their version of Monopoly or their version of Yu-Gi-Oh where they design their cards.



Alex Wong, Justin Oue, Jonathan Chan, Shravan Gajjela, Victor Yu

[mesagames.tech](https://mesagames.tech)

## 16 Poseidon: 3D Humanoid Asset Pipeline

### Team unFYDP



Nathan Jiang, Saswata Gupta,  
David Kang, Hanyu Zhu

Developing 3D graphics is often a necessary step for producing assets in video games and special effects in movies. Within this domain is capturing humans and their movements into the graphical realm, but this process currently requires expensive equipment, complex software, and/or extensive professional work and experience. Such requirements are unreasonable for those new to modeling 3D humans or creating tex-

tures for recreational purposes.

Poseidon is a software pipeline for automatic 3D human modeling based on 2D pictures and videos. All a user needs is a camera. Given pictures of a person from multiple distinct angles, Poseidon is able to generate a textured and rigged 3D model of the subject in the pictures. Given a video of a person (not necessarily the same person as the photos) performing a motion, Poseidon is able to generate a reasonable estimation of the 3D animation. The user can then export the 3D model and animations to be used as they please. Potential uses can include game development, animation, or even further manual enhancements.

The idea of detecting humans and their physical characteristics is a well-researched field within computer vision and machine learning; we take advantage of that to extract information such as model shape (mesh), texture, and limb positions and orientation from the user-provided images and videos. The model texture is stitched together from the input frames, and the contiguous limb positions are turned into an animation. A skeleton is rigged and skinned to the mesh that defines where the limbs are and how they can move. We then apply the animation to the skeleton, and the result is an animated and textured 3D model.

Poseidon is intended to be a toolkit that is easy to set up and use - whether as a casual user's tool or as part of an extensive cloud-based pipeline; the only required inputs are low-fidelity pictures and videos of subjects. The interface then outputs quality assets in an industry standard format (Autodesk FBX). This is a sizable improvement in accessibility and usability for breaking down the high barrier of entry for 3D modeling.

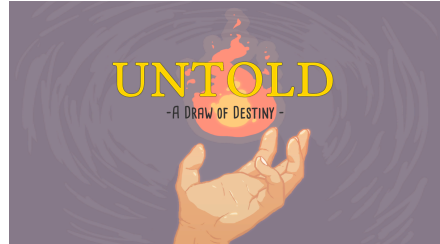
## 17 A card game that uses procedural card generation Team AIGame

Digital collectible card games (CCG), such as Hearthstone and Magic the Gathering: Arena, are a popular genre of strategy games, however the cost and/or time commitment required to keep up with traditional CCGs can deter many more casual players from entering the genre. Our project is a card game that tackles the genre in a new way that can resolve these problems that stem from the card collection aspect. This is achieved by using on-the-fly procedural card generation that aims to preserve the strategic depth of the genre, while minimizing the investment required to keep playing.

Instead of building a deck using the card collection they've gathered, players will pick a deck archetype they want to play. Whenever they would draw a card, the procedural card generation offers them a selection of random cards generated to match the archetype. The different archetype models combined with the card generation logic aim to provide players with cards that operate well together and provide a fun and cohesive experience.

Compared to traditional card games, our approach benefits players, especially casual ones, by requiring less time investment to learn the cards and meta decks. Also, our game eliminates the inequality between decks brought by the need to invest time or money to gather a better card collection.

Our game also competes against draft modes that exist in most CCGs as a deck building mode that also eliminates the requirement of building a collection. However, effective drafting still requires card knowledge, and depending on the skill of the player may still require large amounts of monetary investment. The advantage of our game is that it isn't tied to a set of cards at all thus allowing us to minimize learning requirements and allowing different forms of monetization



Edwin Lin, Mingxuan He, Puneet Singh Ghuman

<https://mx-h.github.io/AIGame/index.html>

## 18 GoSudoku: a real-time, collaborative Sudoku game

### Team GoSudoku



Clive Chan, Corbin McElhaney,  
Mitchell Scott Sandre, Nader Sabahi,  
Samir Rehmtulla, Tilman Lindig

The COVID-19 pandemic caused many people to live their lives primarily online. According to Verizon, online gaming saw the largest share of this shift in the US with a 75% increase in network traffic from gaming.

GoSudoku aims to utilize this growth to combat another COVID-related problem: social isolation. We believe that gaming is a social medium that was relatively unaf-

ected by the pandemic and can therefore fill some of the void left by in-person gatherings. Easy-to-learn, real-time online games serve as a catalyst for people gather (virtually) and have fun together. Our goal is to create a Sudoku platform that enables such socialization.

To make our product accessible to a wide audience, we ensured that existing familiarity with Sudoku is not required to use our product. We chose to make a real-time, collaborative version of the game to best increase socialization. Making it real-time encourages users to play concurrently, likely while in a chat room or video call. Making it collaborative creates a friendlier environment where new and experienced players alike can enjoy games together. The key technical challenges relate to building an interface that is intuitive, responsive, and informative enough to allow for true real-time collaboration among players of all skill levels.

Our platform does not require account creation and does not require a tutorial. The main focus is on creating a seamless interface that encourages interaction outside the app itself. Most online games have much higher barriers to entry, requiring the user to create an account and read a set of rules before playing, that we believe lowers retention for casual players. Since one of our main goals is socialization, accessibility is a key selling point for our product.

We use analytics built into GoSudoku to track the size and engagement of our user base. We hope to see widespread adoption of the cooperative play mode in our product, as well as repeat engagement from social groups on our platform.

<https://gosudoku.io>

## 19 An online multiplayer Mahjong platform Team PlayMahjong

Mahjong is the most popular table-top game in several southeast Asian countries with over 7.6 million playing Mahjong in Japan in 2008. Using colorful tiles to draw and discard until a player has a winning combination of tiles, Mahjong is a game of luck, skill and strategy. Western popularity of Mahjong is growing as more second generation immigrants

learn the game. Despite its popularity, online multiplayer versions of the game are lacking features and inaccessible for English speakers.

Websites for playing online multiplayer Mahjong are not easy to find for a Western audience. Looking for multiplayer Mahjong websites through search engines will turn up few results if any. Websites that do show up are hard to navigate, don't support English users or don't use modern browser-compatible technology.

Our project is to develop an online multiplayer Mahjong platform that will fill this gap in the market, providing a modern online Mahjong experience while also accommodating English speaking users. The platform will be easy to navigate and start playing games, both with friends or with strangers.

Designing a Mahjong web application with comprehensive features presents moderate design challenges. Mahjong's complexity requires fluid UI/UX for users to have a seamless playing experience. Additionally, automated scoring is a non-trivial challenge because of the numerous rules and inherent combinatorial nature of scoring.

Our platform's main advantage is the native support for English speaking users and a modern interface. Currently, the best option for English speaking players is translating other platforms via browser extensions. Our platform will also have advantages over traditional in-person play including automatic scoring, hand analysis and long-distance play without physical pieces.

The core metrics we use to determine success will be usage of the website through number of weekly active users, unique users, site visits, and games played. We hope to be the primary website to play Mahjong and be easily accessible for everyone.



Aiden Benner, Daniel Tong, Derek Seo, Jeffrey Xiao, Matthew Ban Kim, TJ Yang

[playmahjong.io](http://playmahjong.io)

## 20 Yada: Yet Another Docs App Team Burrows



Akshay Kumar Pall, Carl Shen, Shehan Suresh, Yashashvi Aravind Segu

When forming ideas, humans rarely think sequentially, going from abstract concept to explanation to presentation. Instead, cluster thinking is commonly used, which involves approaching ideas from all of these fronts, and more, concurrently. Unfortunately, the writing tools we use today do not facilitate this way of thinking and are limiting our potential.

The objective is to build a platform that allows people (writers, creators, engineers, etc.) to add their ideas to a knowledge base and later organize those ideas into concrete views. The platform should eliminate the need to maintain duplicate information across different documents by maintaining consistency between the knowledge base and the views that are created from it.

For this project we built Yada. Yada is a platform for note-taking, document creation, and knowledge aggregation, where individual lines/blocks can be annotated/tagged. These blocks can be used to generate views that maintain consistency with any updates made to the source document. Supported views include flash cards, slideshows, and text documents, which can be used for various purposes such as studying, presenting, project planning, and more. In addition to view generation, tags provide a novel way to search text, which makes knowledge navigation and organization a lot easier.

Yada competes with other note-taking apps such as Google Docs and Notion, but neither of them offers the paradigm of a single source of truth document being used to generate views. The ability to tag blocks within a knowledge base and use them in other documents is a novel idea that takes advantage of the way humans normally think and create.

We are building Yada as a new product. Ultimately, we achieve our goal by gaining users that can effectively take advantage of our platform and improve their workflows. We believe that Yada can give users the power to replace the mess of written content they have with a more natural model of idea creation and presentation.

<https://yada.dev>

## 21 Syllable: Say everyone's name (correctly) Team Pronounce

People's names express cultural background, values, and identity. The majority of the English-speaking population in countries like Canada and America has an English name. As immigrants add to the cultural diversity of our society, there have been much more significant variations in names than before. Non-English names such as Chinese, Polish, and Arabic ones can all be challenging for English-speaking people. Moreover, non-English speakers may have also struggled to pronounce English names, and people with uncommon names often give up on constantly correcting others' pronunciation. Mispronouncing a name may cause embarrassing situations and negatively impact people's socio-emotional well-being.

Believing that saying people's names properly is crucial in any social interaction, the mobile app Syllable reduces the friction and frustration in this step. People from the same company or the same class can record their names and upload them to the app, and then other people can listen and practice. User' practice pronunciation can be evaluated manually by the original recorder, so they will know whether they need to practice more.

Research suggests that people with "foreign" names are more likely to be marginalized in various ways. Thus, we expect Syllable to promote social awareness of ethnic equality, respect, and trust. During some initial stage user research, the app showed it could be helpful in both academic and professional contexts. For example, during the university convocation ceremony, faculty directors need to say everyone's name. With Syllable, they can confidently pronounce the names correctly in such an important moment in students' lives.

An MVP implementation is completed. In the last term, several exploratory user activities was conducted to align the project direction with the needs of potential users. Right now, the team is preparing to launch its first beta testing during the convocation in summer 2021.



Jarvis Wu



## 22 Data-driven product comparison website Team Pronto



George Eisa, Manuel Dionne, Sam Alberico, Shane Hickman, Wilson Wu

Over 85% of consumers conduct online research before making a purchase online. To help inform consumers, many websites publish product reviews, but this is normally a manual and time consuming process. Another shortcoming of this system is that consumers must visit multiple websites to see diverse opinions. Through automating product review generation with data collected from many sources, our website Comparisum has a competitive cost advantage,

and produces reviews which summarize the collective opinion of many users.

The primary function of Comparisum is to generate data-driven reviews of products, focusing on common complaints and praises. Once generated, reviews are displayed for individual products and as comparisons between competing products. Each review includes images, current prices at different retailers, a short description, technical specifications, a summarization of pros and cons, and the prevalence of each pro and con among all reviewers. When users follow affiliate links and purchase the products through our retail partners, Comparisum receives a commission.

The data for review generation is retrieved via web crawling. Using machine learning, the sourced data is aggregated into lists of pros and cons for key product features, and an overall score is calculated based on user sentiment. This is combined with information from retail partners to formulate a full review. Current trends inform the choice of which products to review and compare.

Challenges include crawling data, reliably categorizing unlabeled data, abstractive text summarization, and unique product identification; we propose several novel solutions for tackling all these problems.

Comparisum's success will be evaluated on total website traffic, affiliate link conversion rate, and ultimately overall affiliate commission revenue. We believe that by automatically generating unbiased, high quality reviews at a low operating cost, we can grow to become a reputable review and comparison site.

<https://comparisum.com>




## Student Index

Aiden Benner, 19  
Akshay Kumar Pall, 20  
Alex Wong, 15  
Aman Arora, 12  
Ambareesh Balaji, 1  
Andrew Lamb, 6  
Armanit Garg, 11  
  
Ben Upenieks, 9  
Beverly Vaz, 8  
Bill Qi, 11  
Bo Yuan Tan, 7  
Bohan Shi, 10  
  
Carl Shen, 20  
Carmen Kwan, 13  
Cathy Wang, 13  
Chaitanya Varier, 9  
Chengzhong Meng, 9  
Chris Pardy, 12  
Clair Hu, 10  
Clive Chan, 18  
Corbin McElhanney, 18  
Curtis Phan, 9  
  
Daniel Tong, 19  
Daniel Xu, 10  
David Kang, 16  
Declan Goncalves, 1  
Derek Seo, 19  
Dhruv Jauhar, 12  
Doris Zhang, 7  
Doug Sawadsky, 14  
  
Edwin Lin, 17  
Emily Lin, 8  
Ethan Xue, 11  
  
Frank Kang, 10

Cover art by  
Kathleen Wang

Frank Liu, 11  
  
George Eisa, 22  
Guanzhong Chen, 14  
  
Hanyu Zhu, 16  
  
Jack Williamson, 9  
Jarvis Wu, 21  
Jason Xu, 4  
Jaxon Lin, 2  
Jeffrey Xiao, 19  
Jenny Wills, 12  
Jerome Alve, 3  
Jiayi Zhou, 6  
John Weber, 3  
Jonathan Chan, 15  
Judy Liu, 5  
Justin Oue, 15  
  
Kalvin Thye, 5  
Kody Tang, 5  
Krisztian Palvolgyi, 4  
  
Logan Mosier, 12  
  
Manuel Dionne, 22  
Matthew Ban Kim, 19  
Mengfei Liu, 7  
Michael Cai, 11  
Michael Yaworski, 14  
Michael Leng, 6  
Milan DiGiuseppe, 4  
Mingxuan He, 17  
Mitchell Scott Sandre, 18  
  
Nader Sabahi, 18  
Nathan Jiang, 16  
Nathan Ko, 2  
Nicholas Good, 6



Nicholas Throm, 14  
Nick Geofroy, 9

Paul Janiszewski, 13  
Philip Scott, 1  
Puneet Singh Ghuman, 17

Rachel Lang, 7  
Roxane Fruytier, 12  
Ruiying Bu, 8  
Run Tong Li, 14

Sam Alberico, 22  
Samir Rehmtulla, 18  
Sarah MacLean, 3  
Saswata Gupta, 16  
Shane Hickman, 22  
Shehan Suresh, 20  
Shela Qiu, 13  
Shravan Gajjela, 15

Tilman Lindig, 18  
TJ Yang, 19

Victor Yu, 15  
Vincent-Olivier Roch, 9

Will Brown, 4  
Wilson Wu, 22

Xiao Zhou, 10

Yashashvi Aravind Segu, 20  
Yingjie Guo, 8  
Yixuan Wang, 2







UNIVERSITY OF  
**WATERLOO**



FACULTY OF MATHEMATICS  
**DAVID R. CHERITON SCHOOL  
OF COMPUTER SCIENCE**

FACULTY OF ENGINEERING  
**DEPARTMENT OF ELECTRICAL  
AND COMPUTER ENGINEERING**