



bloxeo

Project Deck

# Table of Contents

1 Introduction

2 Problem

3 Design Research

5 User Research

7 Audience

12 User Flow

13 Wireframing

14 Design Comps

16 Dev Data

24 Thank You

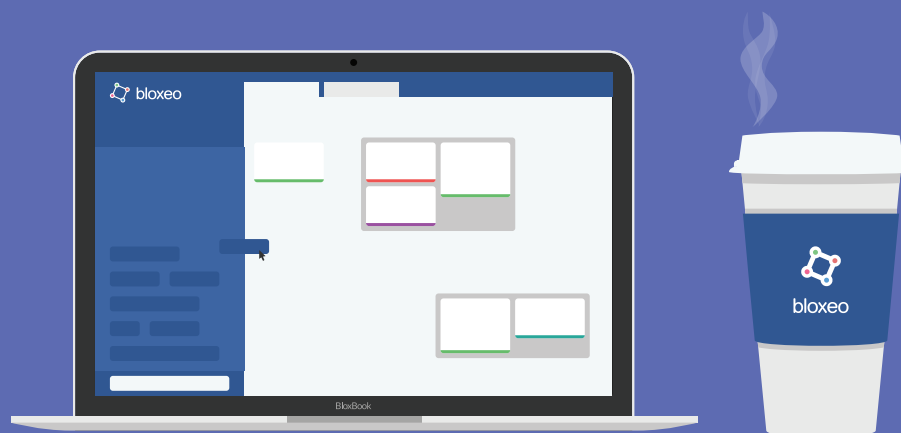
# What is Bloxeo?

## Product Description

Bloxeo is a web-based collaborative brainstorming application that creates an experience making your creative process not only easier and more fun, but more productive.

## Top Features

- Assemble a team and work off of each other's ideas.
- Generate ideas into a growing bank of "idea blocks."
- Sort through ideas, and build on the best ones.



# Let's make **your Eureka moment** happen.

## Concept

Brainstorming is hard. How can we make it easier to organize and build on our ideas? By maintaining creative pressure within a relaxing and playful environment, people can produce their best ideas.

## Goals

- Create a platform for organizing ideas with a team
- Build a tool for creative minds to use when brainstorming
- Design an experience that helps creators refine concepts

# Playful, free, and light.

## Moodboard

We wanted to create a **light** yet energizing environment that promoted **creativity** and **productivity**.



# Easy to use, fun to use.

## Feature Board

We wanted our interface to be as **intuitive** and **fun to use** as possible. This meant looking into other **successful** interfaces.



# User Research

We **straight-up** asked **what users want**.

Before the design and development of our application, we went beyond ourselves and surveyed many people of various professional backgrounds. We wanted to know what kind of tools they use and what kind of tool they need.



# What did they tell us?

58%

said that they have used an online tool for brainstorming.

## Top Tools

- Facebook chat
- Google Drive
- Trello
- GroupMe

91%

found brainstorming frustrating at times.

## Of these people,

14%  
said the most frustrating part was lack of contribution

12%  
said the worst part was coming to a compromise

91%

said they usually worked in groups of 5 people or less.

## This told us

- People usually work in smaller groups when they brainstorm.
- 1-5 people is an optimal group size for successful brainstorming.



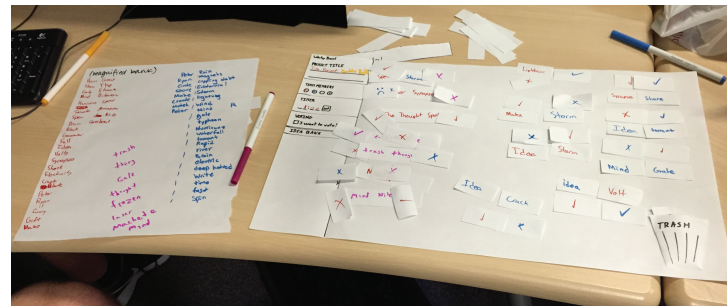
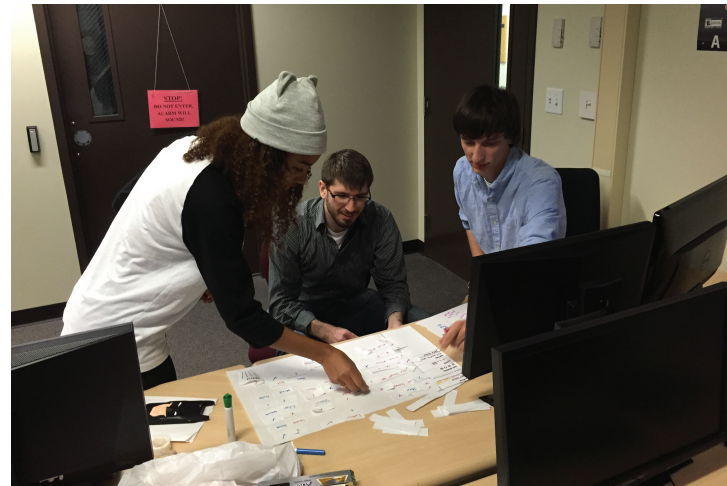
# Paper Prototyping

## User Testing - the fun way.

We created paper versions of our application and had several groups of 2-3 people try to use them. We then took feedback from them to improve our design.

## Feedback Recieved

- Users wanted the ability to add new ideas after the timer had run out.
- Although paper “idea blocks” were provided in different sizes, users only used one size.
- Some users found that the order of their grouped blocks mattered to them.



# Who is our audience?

Primarily **college students** and **young professionals** working in small teams to achieve a common goal.



Age: 18-28

Gender: All

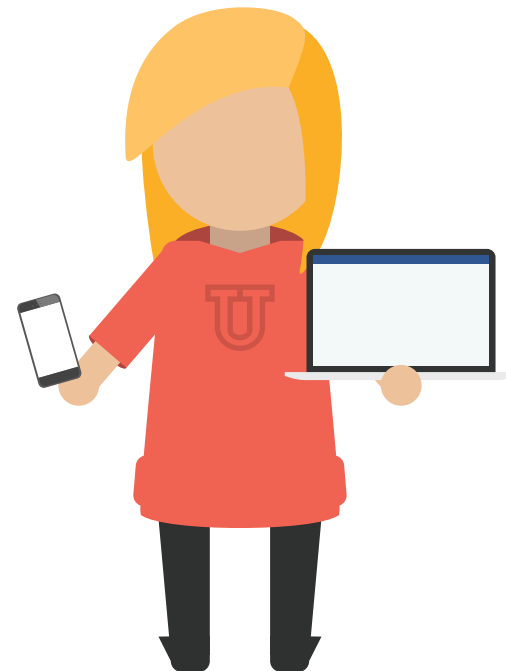
Creative  
People

# Amelia Rose

Design Student | 18 years old

Amelia is a design student at a US university. She works on a variety of different projects for her classes and does have experience designing for web/mobile. She is pretty well acquainted with technology and uses her iPhone for most of her communication needs.

Since most of her projects are group-based for classes, they tend to have to work on various students schedules. Amelia does very well in school and is very organized in how she keeps track of her project due dates. However having to work with students who aren't always like her, she often has to worry about them slacking.

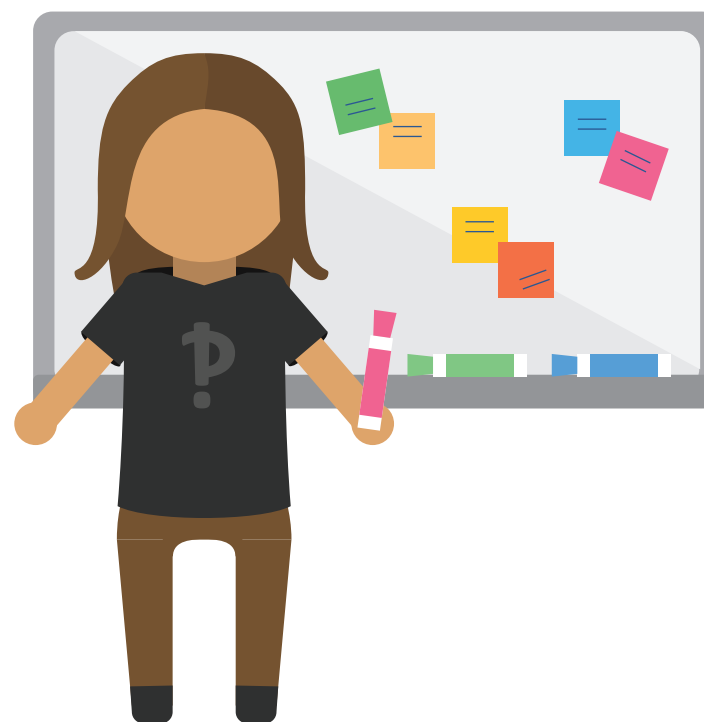


# Kate Defeo

Creative Designer | 23 years old

Kate works as a designer at a small web and mobile start up. She lives in a city with a strong art community. She pulls a lot of inspiration from art festivals and other creative events and people attending. However, she still makes it a priority to keep up with design news from a few different sites to see the new trends in the industry.

She and her small team often spend a lot time in meetings with clients to go over their goals. When not in meetings, they spend their time brainstorming in the “idea labs” her office has. Being a visual person, she enjoys sketching and using the post it method so that she can visually categorize ideas as she thinks of them.



# Jerry Cosworth

Game Programmer | 28 years old

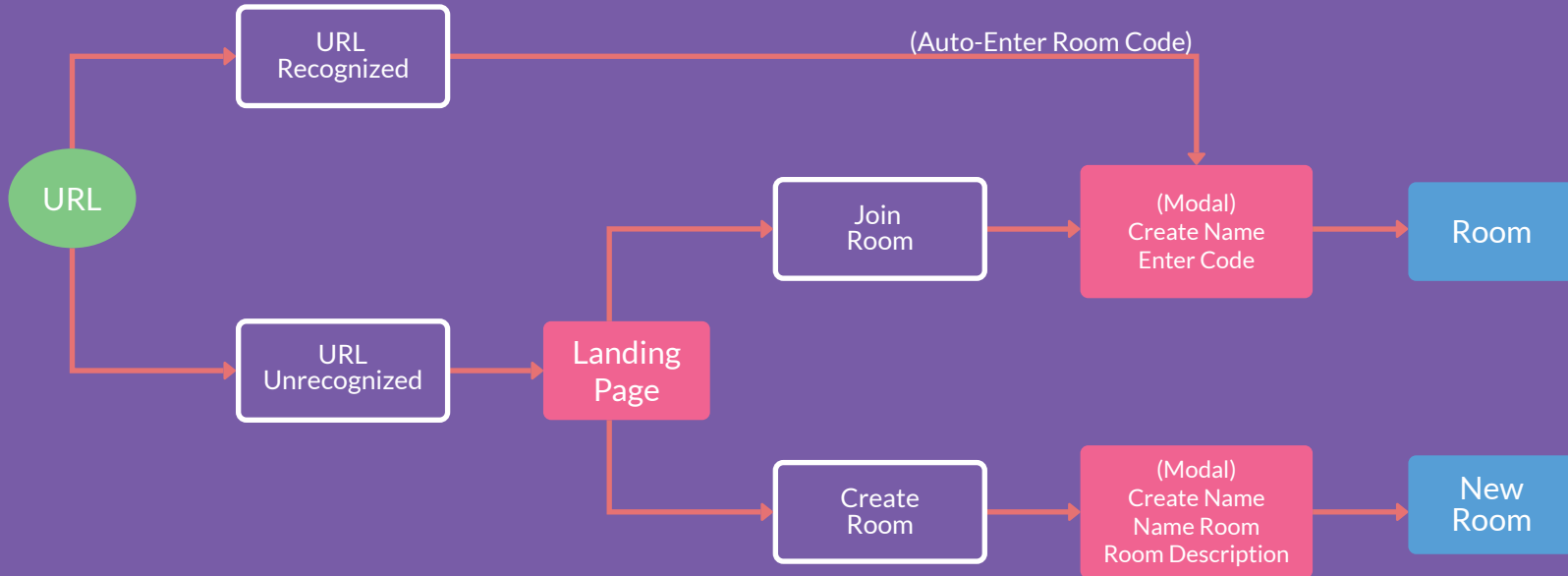
Jerry is a game programmer at a AAA game studio. He enjoys using github to keep track of the projects he works on and typically just uses google docs for his documentation due its ease and accessibility. When he isn't programming, he's probably gaming.

To improve his skills and meet new people, Jerry participates in game jams and app jams regularly. During those events, he is given 48 hours to create a fun app or game with a small team that follows a theme. They usually pull inspiration from their favorite movies or comics. Due to the nature of the jam, they have to be pretty rigid in what the requirements and scope of the project will be. However, this doesn't always stop them from struggling to compromise on what those features will be.



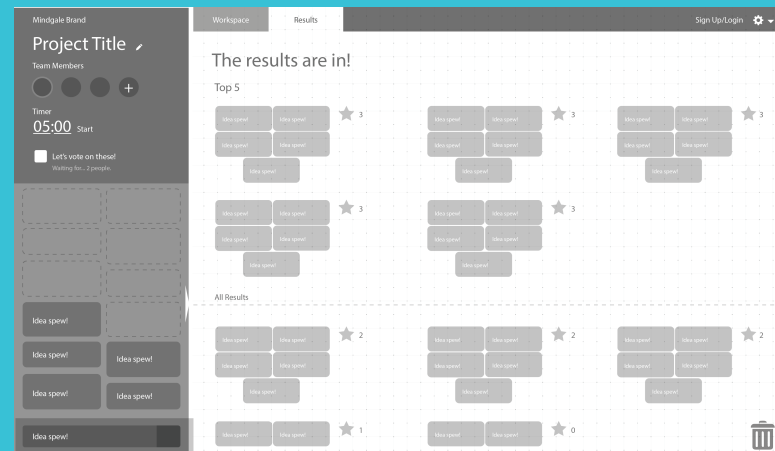
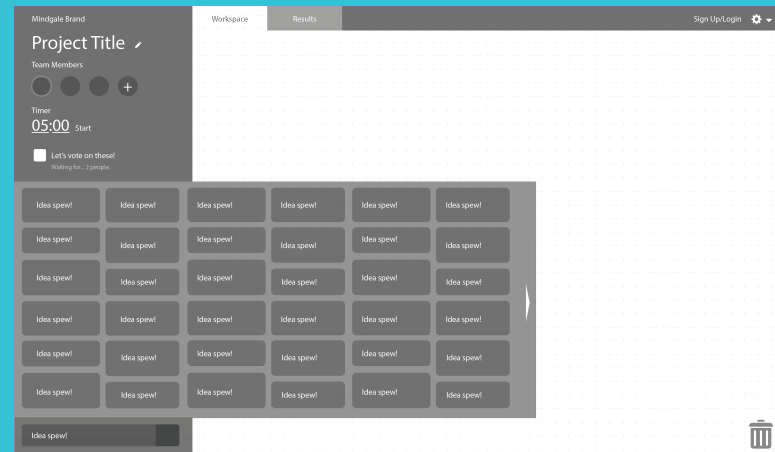
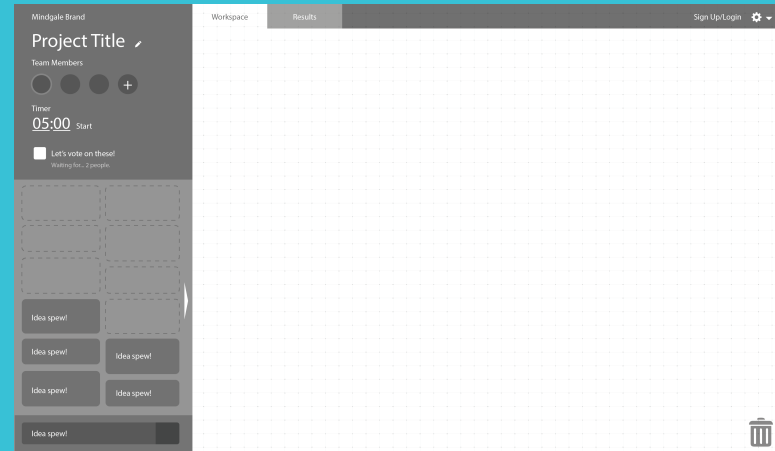
# The Flow

We made it easy for **anyone** to **jump right in**.



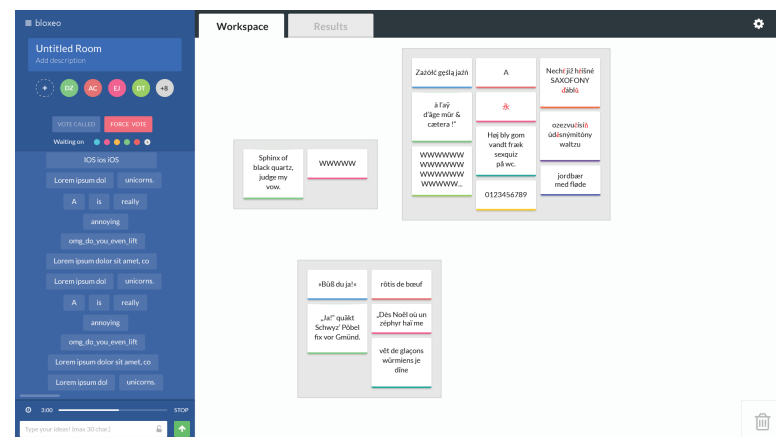
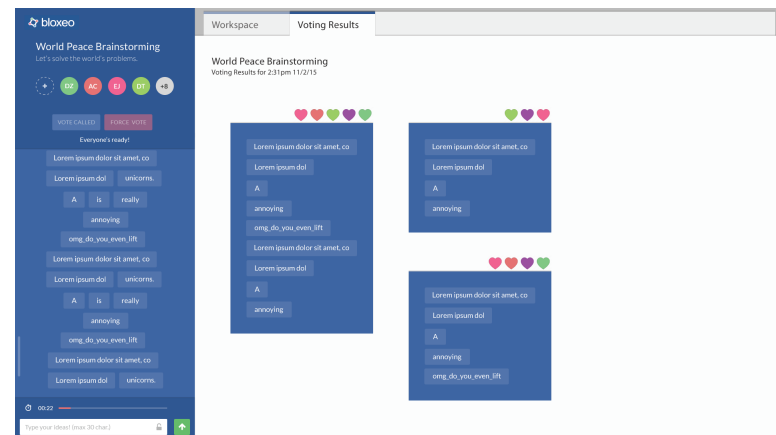
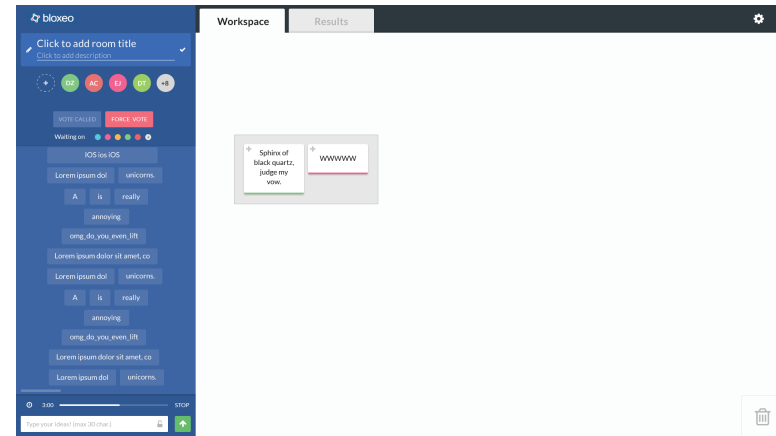
# Wireframes

The initial wireframes focused on clean, open interface with various features incorporated such as a timer, expanding idea bank, and results tabs. All of these features made it to the final design but looking and functioning different ways.



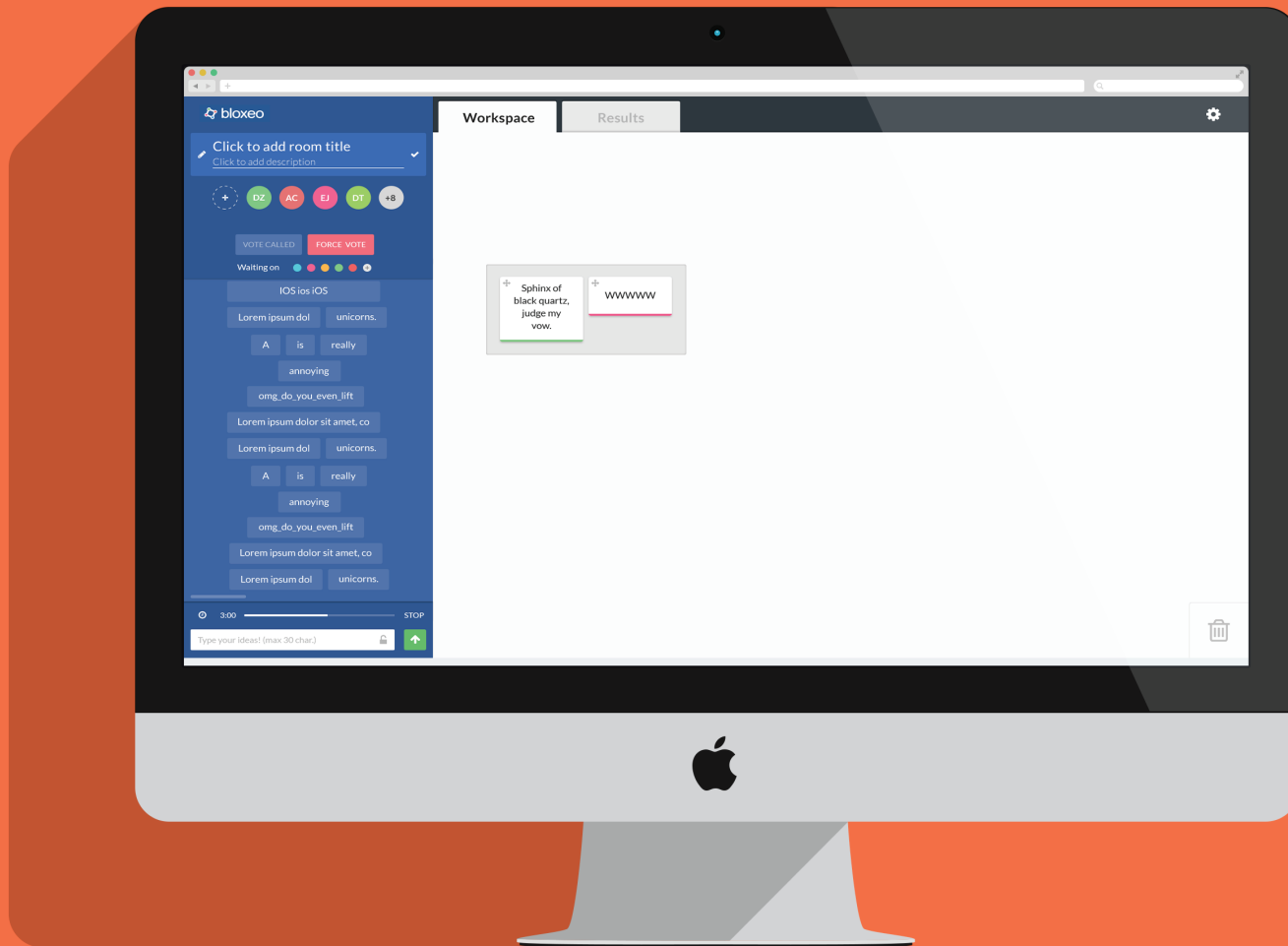
# Design Compositions

By combining the user research and the wireframes, we were able to make compositions that reflected how Bloxeo functions.





# Final Composition



# Branding Guidelines

## How to treat our logo:

- The full-color version of the logo should only be used on Bloxeo blue.
- Keep enough space around the assets so they appear clean and uncluttered.
- Don't modify Bloxeo brand assets, such as by changing the design or color.
- Please only use the logo variations presented here to represent Bloxeo!

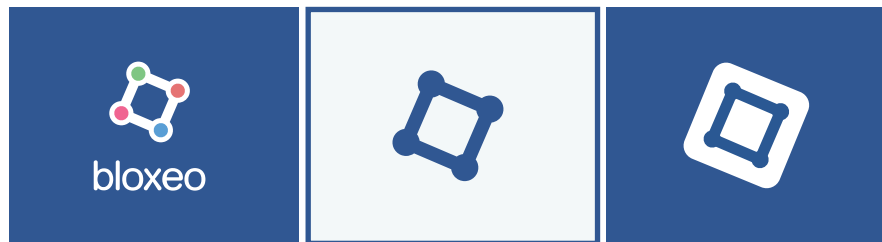
## Logo & Watermark



## Monochrome Logo & Watermark



## Logo Variants



Vertical

One-Color

Monochrome

# Visual Identity

Being a **collaboration-based** app, **Bloxeo** needed a clean interface for a **smooth user experience**.

By choosing a simple typeface and primary color palette, we assist without being obtrusive.

However, our secondary color palette is multifaceted to help users easily distinguish whose ideas are whose. As a user of Bloxeo, you can expect to be assigned any of the vibrant colors on the right.

## Typeface

### Lato

AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz

## Primary Color Palette



primary blue  
#34578f

secondary blue  
#41659f

background blue  
#2e373d

primary light grey  
#fbfbfb

secondary light grey  
#d7d7d7

## Secondary Color Palette



red  
#EF5350

pink  
#EC407A

purple  
#AB47BC

deep purple  
#7E57C2

indigo  
#5C6BC0



blue  
#212121

light blue  
#29B6F6

cyan  
#26C6DA

teal  
#26A69A

green  
#66BB6A



light green  
#9CCC65

lime  
#D4E157

amber  
#FFCA28

orange  
#FFA726

deep orange  
#FF7043

# Developers Rock

Our **amazing team** of developers made sure our users got a **smooth** and **enjoyable experience**.



# Technologies of Choice

## Client SPA

React & D3

[View](#)

Flux

One-directional data flow  
via AJAX & Websockets

## Server

Express RESTful HTTP

Socket.io Websockets

ioredis / Redis

key/val store

Mongoose/Monyo

Document Store

Node

Runtime

Heroku

Host

# Client Data

## Write Up

We used React.js/Flux to structure our front end MVC, with Socket.io for websocket communication. Babel was used for advanced javascript technologies, and Grunt for Dev task automation.

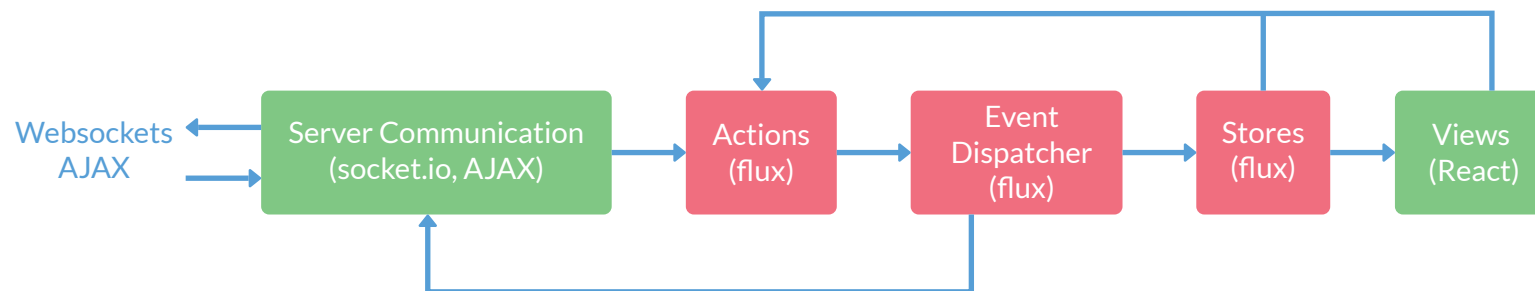
## Why React over Angular?

When choosing a framework we decided to go with React.js over Angular.js. This came down to React having an easier learning curve and Angular going through huge changes to the API that would cause future development issues.

# Client Architecture

## Front End Stack

- React.js/Flux - MVC
- Websockets(socket.io) - server communication
- Babel - Allows us to write es6 compatible js
- Grunt - task automation



# Server Data

## Write Up

For relational data we used MongoDB and everything else we put in Redis. All communication that occurs in a workspace needed to be as near-realtime as possible so we used WebSockets, but for everything else we took advantage of the added structure of a RESTful HTTP API.

## Why Express over Sails?

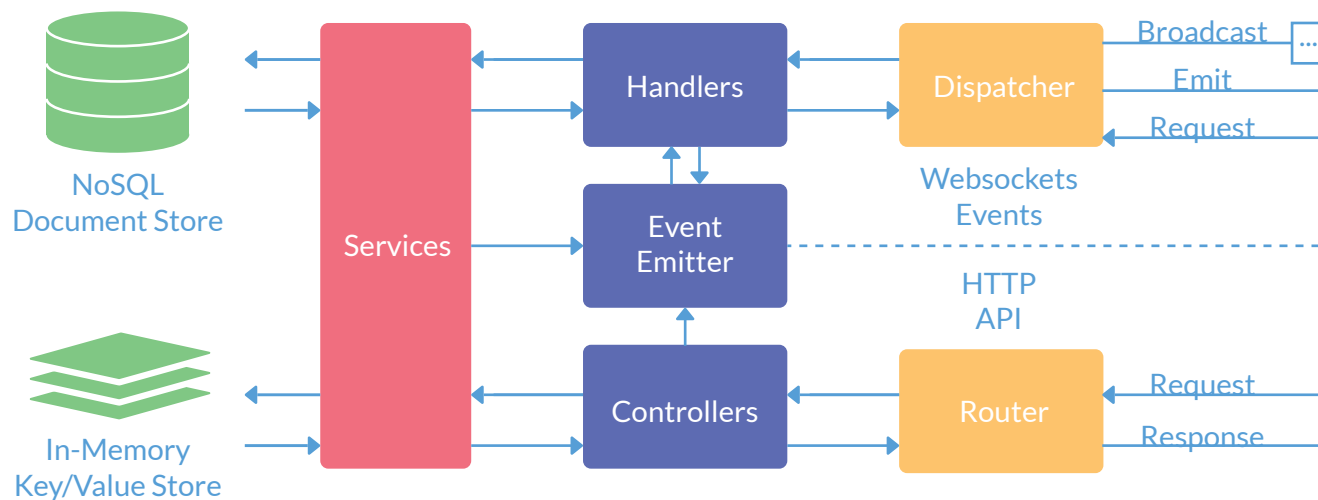
Initially this was a Sails project and it was going well, but there were definite growing pains. Their ORM, Waterline, was slow and buggy, their Socket.io wrapper was out of date, and Sails itself relied on an old version of Express. The main takeaway for us was that composable libraries are better than end-to-end frameworks.



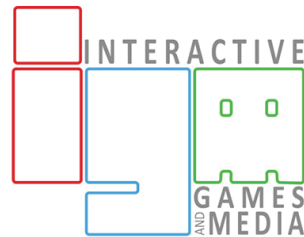
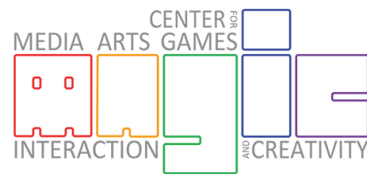
# Server Architecture

## Back End Stack

- Websockets (Socket.io) and HTTP (Express) - for server communication
- Mongo (document database) and Redis (in memory key/value store) - persisting data
- Node.js runtime on Heroku platform



# Made possible by:



## And an amazing team:

### Leaders

**Cody Van de Mark**  
Douglas Mansell

### Designers

**Jason Peretz**  
Anthony Amici  
Kathleen de la Fuente  
Ryan Hirsh  
Rosetta Lediard  
Daniel Zhu

### Front-End Devs

**Peter Gyory**  
Cailin Li  
Ashton Wai  
Dustin Raimondi  
Ryan Ferrell  
Stephanie Jurgiel  
Trey Borro

### Back-End Devs

**Will Paul**  
Braxton Frederick  
Eric Kipnis  
Nicholas Minnoe  
Isaiah Smith  
Chad Karon