

# Schedgy Project Blueprint

Will Thorbecke and Jeremy Garcia | FSE Spring 2018

## Project Abstract

Schedgy is a gamified calendar app that allows user to input tasks they have to do -- time estimates and difficulty ratings related to those tasks -- and it'll automatically input these tasks into the calendar based on when the user is free and when the user is most efficient. Each task gives the user a certain score and introduces a gamification element, which is further accentuated by the app's assistant, Schedgy -- an interactive character that gains experience points by completing tasks.

## Goals

Team goals:

1. Follow what little design principles we have to make an app that is user friendly and visually appeasing
2. Maintain a professional codebase and follow good practices relevant to the language/framework we're using
3. Get gud at Go
4. Show a little hustle

Will's goals:

1. Learn to write better tests in the spirit of creating an app that is rock solid and production ready
2. Learn more about web-dev fundamentals (i.e., be able to argue why one framework should be used over another)
3. Become comfortable at Go and React
4. Learn a little about what design practices make a webapp look visually appeasing

Jeremy's goals:

1. I want to be able to create a webapp that is production-ready.
2. Learn Go and Sql
3. Make the app scalable (e.g. easily add Redux if we want)
4. Write good tests

## Feature Set

### *MVP Features*

- Functioning calendar
- Feature that fills up your calendar with things that the user inputs.

### *Next Steps*

- Reward system for completing tasks in the form of visual stimulation
  - Have users give due dates and proposed difficulty for items on to-do list

- Gamify the calendar/scheduler
- Create a deity/wide sage that spits out what you should do
- Introduce a ML/algorithmic element that takes into account when a user is most efficient or does work at all, and then proposes tasks at that time period
- Be able to team up with friends to do common tasks
- Gmail/Outlook calendar syncing

## System Architecture

Database: mySql

Backend: Golang

Frontend: ReactJS

## Project Timeline

- |           |   |
|-----------|---|
| Weeks 1-3 | <ul style="list-style-type: none"> <li>• Set up project</li> <li>• Create a (very) basic calendar app</li> </ul>                                |
| Weeks 4-6 | <ul style="list-style-type: none"> <li>• Work on combining algorithm</li> <li>• Start working on Schedgy</li> <li>• Work on Accounts</li> </ul> |
| Weeks 7-9 | <ul style="list-style-type: none"> <li>• ...</li> </ul>   |

## Risks

1. Task and Calendar combining algorithm becomes very hard
2. Gamification aspects (e.g., animations and art) end up consuming all our time
3. Might hit a roadblock with Google calendar api if we end up syncing our app with that
4. Codebase becomes very messy and unprofessional
5. Code quality loses value over time, best practices won't always be implemented