

Team FlaskChained
SoftDev1 pd<1>
P2 -- The End
2020-01-05

INITIAL DESIGN DOCUMENT

Jesse Hall (*PM*)
Ayham Alnasser
Hong Wei Chen
Ivan Galakhov

Project Sudocool

PROGRAM OVERVIEW:

The classic logic game, Sudoku, in a portable website built with JavaScript. The puzzles are automatically generated in accordance with difficulty settings: easy, medium, hard, killer. Users can create an account to save finished puzzles and times or can play without an account. A set of weekly leaderboards is displayed on the page, ranking the users by the amount of time it took for them to finish the puzzle, differentiated by the difficulty setting. If the user is stuck, they can request hints, with a certain penalty being added to their time setting.

PROGRAM COMPONENTS:

1. Sudoku Solver

Some of the simpler puzzles can be solved in negligible time, but solving the killer puzzles may be costly to the efficiency and enjoyability of the site. We may need to rehash the algorithm discussed in Mr. Brooks' AI course to make it more efficient or rewrite a solver entirely. We could build a database of the puzzles and their solutions upon generating each one, and then have the site call on the DB for the puzzle if a user requests one.

2. Puzzle Generator

Generating unique puzzles involves a pseudorandom number generator that creates solvable, but also difficult, puzzles. Difficult puzzles will need to have other characteristics aside from simply giving the user fewer numbers, otherwise, it would be a frustrating experience and wouldn't test their skills. One way to test how hard a puzzle is to run it with the solver, and see how many times the solver backtracks while working on the puzzle. The more backtracks there are, the harder the puzzle is.

3. Front-end

We anticipate that the hardest part of this project will be to make a front-end interface that allows users to interact with the sudoku puzzles. The number input area would need to be conducive to the sudoku experience, i.e. allowing the user to fill a box with all possible numbers that are legal and removing them as they find other ideas. A website that only uses basic HTML forms would be arduous and would be infuriating to the user. So, we expect a great deal of JavaScript to be involved with our project. Besides that, our FEF is going to be Bootstrap, as that is the framework most of our group is familiar with.

4. Areas of Expansion

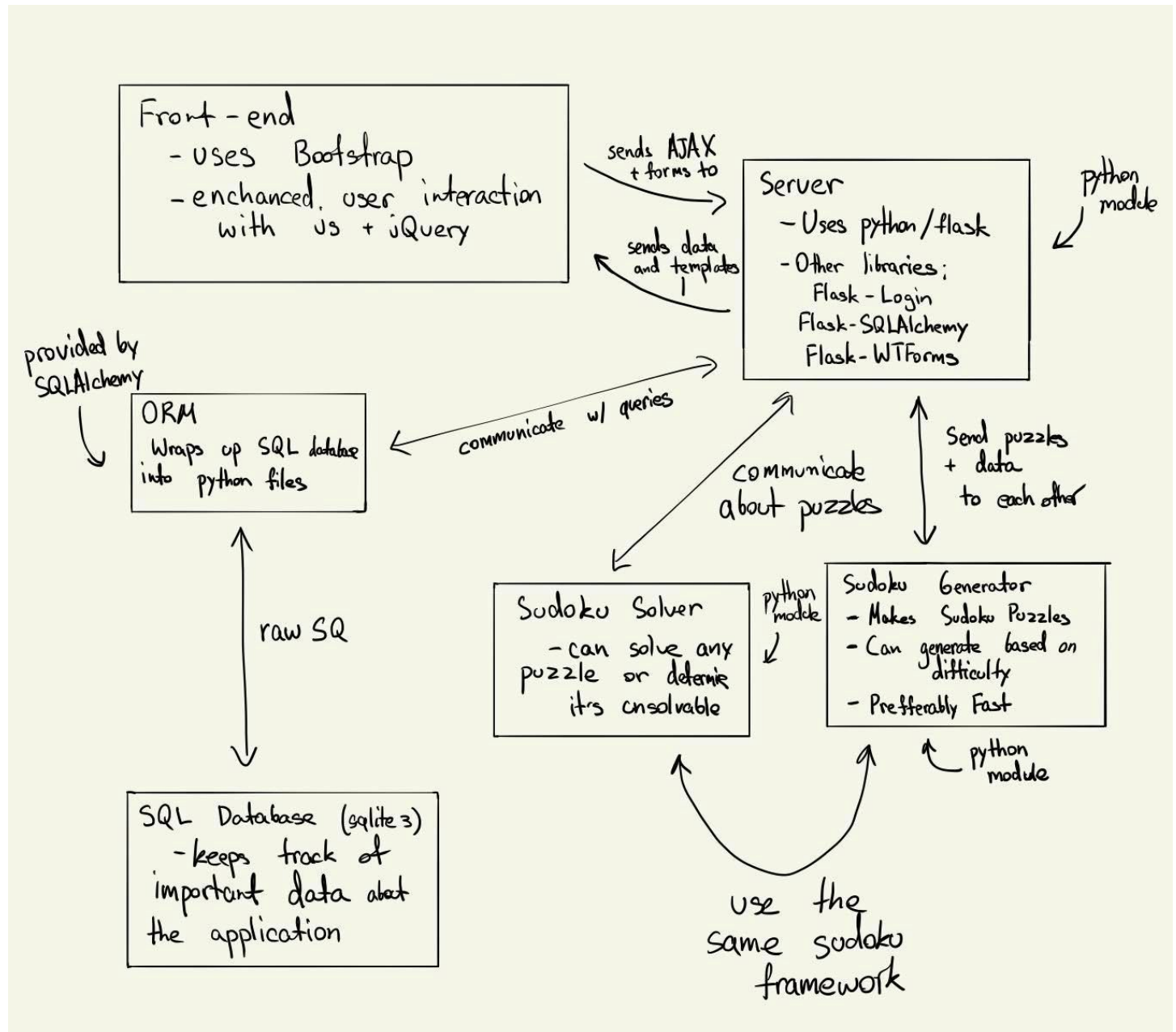
Besides the effective user interface for solving puzzles, the MVP for this project is fairly low. For this reason, we propose the following areas of expansion, assuming we finish our MVP early:

1. Some kind of way to save puzzles while working on them
2. Allow users to create their own puzzles (and make sure they're solvable)
3. PDF Puzzle Export
4. More types of puzzles (ex. Symmetric Sudoku)

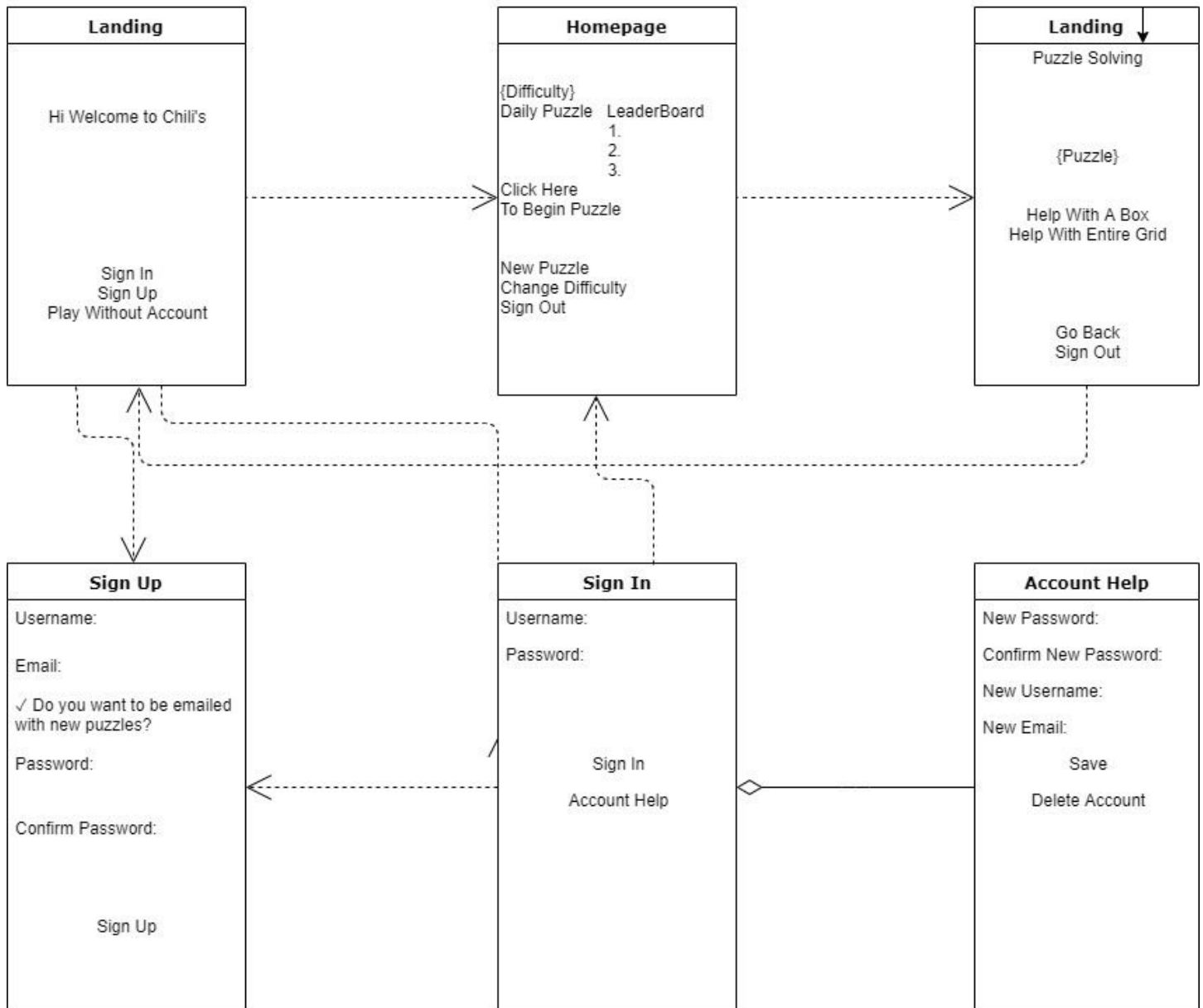
ROLES

1. Frontend Dev including JS (Ivan and Jesse)
2. DB Management and Design (Ivan)
3. User Accounts and log in (Ivan)
4. Sudoku Solver and Generator (Ayham)
5. Email Management (Ayham)
6. Frontend Design including CSS (Hong Wei and Ayham)
7. Leaderboards (Hong Wei)

COMPONENT MAP



SITE MAP



DATABASE MAP

