

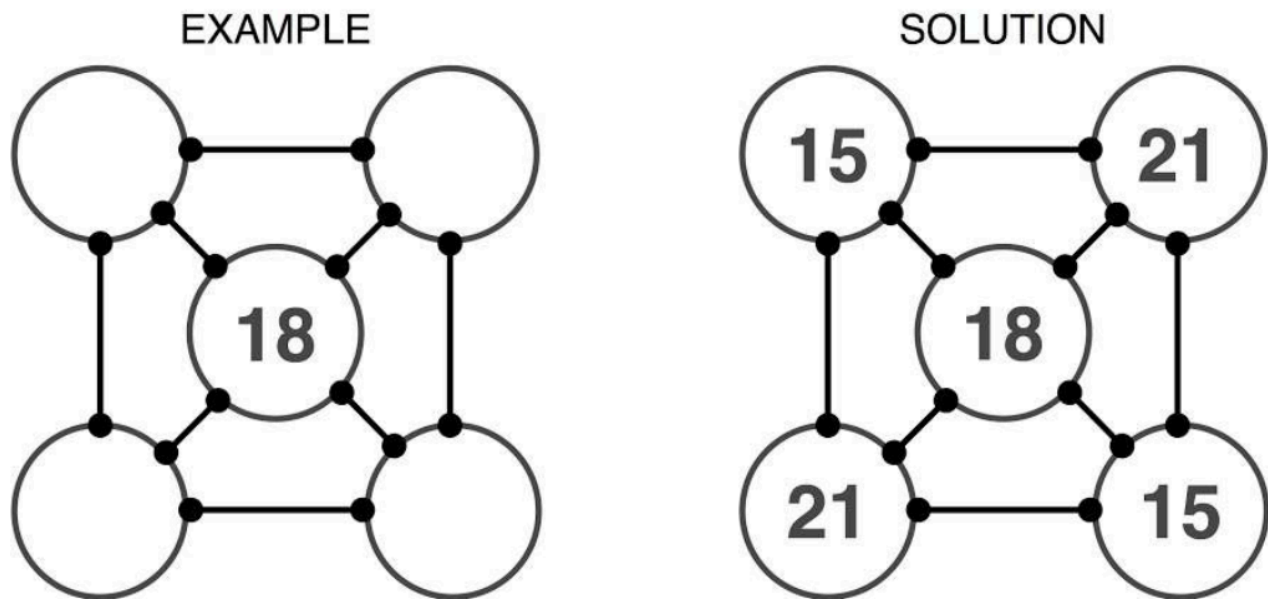
Product Design Specification

Project Description

Detailed Description of the End Product

([Source](#)) The user is given a diagram of interconnected nodes. One node will have a number in it. The user must fill the other nodes with positive whole numbers in such a way that each node's number is the sum of the digits of all the numbers connected to it.

Ex:



The solution works because:

$15 = (2 + 1) + (1 + 8) + (2 + 1)$ for the 2 corners

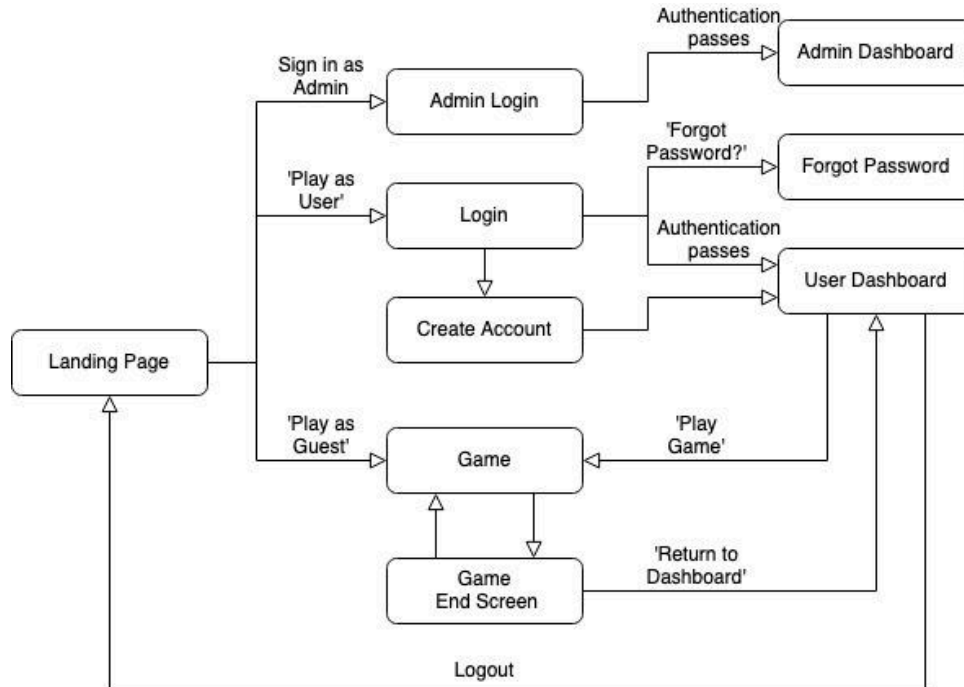
$21 = (1 + 5) + (1 + 8) + (1 + 5)$ for the other 2 corners

$18 = (1 + 5) + (2 + 1) + (1 + 5) + (2 + 1)$ in the center

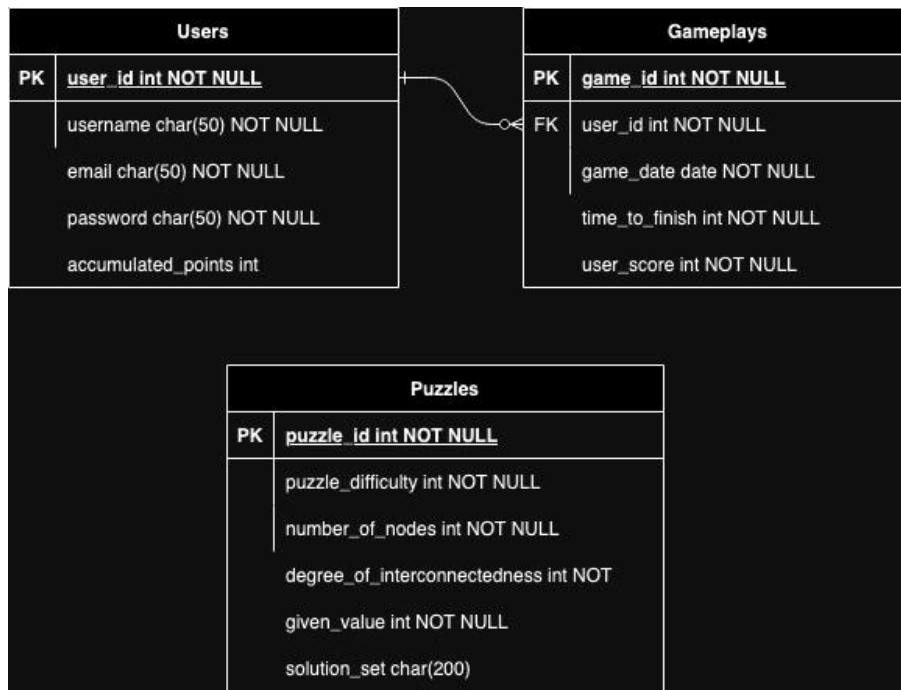
- + **Gameplay Mechanisms:** Users will be presented with a diagram of interconnected nodes. When a user clicks on a node, they will be able to type in a number for their guess. After they have filled in the nodes, they will click the 'Check Answer' button, which will see if the provided solution is acceptable. If it is incorrect, users will be alerted that their solution is incorrect and will be provided with a hint as to which nodes are incorrect. If it is correct, users will be greeted with the game end screen, which will display the total number of points they've earned, as well as options to either continue to the next puzzle or to go to the user dashboard.
- + **Hint Mechanisms:** Users can press a 'Get hint' button while they are playing the game, which will result in a popup that suggests a number they can use to fill in a particular node. If they get really stuck, users can also press a 'Get solution button' and get one of the possible solutions for the puzzle (though, they will not receive any points if they press this button).

- + **Point Mechanisms:** Users will gain points based on whether their solution is correct, the time it takes to complete a puzzle, how many nodes there are, and how many nodes are interconnected to each node.

User Flow Design Diagram



ERD



Methods Used in the Puzzle Solver

- + **Generating Answers:** A backtracking algorithm will be used to generate all the possible answers for a given diagram. For each connected node that hasn't been assigned a value yet, a positive whole number will be assigned to it. All possible assignments will be recursively explored and made sure that the sum equals the value of the node it's connected to. If a solution is found where all filled with valid numbers, it will be added to a list of possible solutions. Other possible assignments will be tried after backtracking, until all options are exhausted.

Market Space and Selling Points

The game is geared towards people of all ages of any kind. However, math-oriented people and puzzle-lovers are more likely to consistently engage with the game over time.

The game is simple enough in the sense that the directions are very straightforward and there's not much the user has to do, but still challenging enough for users to be engaged. There will be many different puzzles, so the game will not become redundant and users can progress onto more difficult problems if they want something more challenging.

With this game, users can flex their problem solving and logic skills, as well as practicing their math skills through the constant additions needed to solve the problems.

Functional Specifications

List of Features

- + **Landing page:** When users land on the site, they'll be greeted by the home screen. There is a navigation menu at the top where users have the option to log in, sign up, or go to the game. Users will have the option to 'Play Now!'.
- + **Account creation:** Users will be able to create an account. With an account, users will have their game history recorded.
- + **Password hash/encryption:** The user's password will be encrypted before it is stored in the database to make sure the user's information will be safe in the case of hacking.
- + **Login:** If users have already created an account, users can login to the site with their email and password to play a game.
- + **User authentication:** The user's identity will be verified after they login and before they move on to the rest of the site. If authentication fails, the user will be asked to enter their information again.
- + **Logout:** When a user is finished playing, they can logout of the site.
- + **Game Menu:** The user can access a game menu, where they can choose from many possible puzzles to complete.
- + **Game:** The user is given a diagram of interconnected nodes. One node will have a number in it. The user must fill in the other nodes. They will be able to type in positive whole numbers in each node such that each node's number is the sum of the digits of all the numbers connected to it.
- + **Submit Solution:** When users are finished with their graph solution, they can press a 'Submit Solution' button to see if their solution is valid.

- + ***Get Solution:*** If users get really stuck while trying to solve a puzzle, they can press a ‘Get solution button’ and get the solution for the puzzle.
- + ***Game end screen:*** After a user successfully completes a level, they will see a screen displaying how long it took them to complete the puzzle, as well as options to either continue to the next puzzle or to go to the landing page.
- + ***User dashboard:*** Users will have a dashboard where they will be able to view their account settings and gameplays.
 - + ***Account settings:*** Users can change their name, email, and/or password through account settings on their user dashboard.
 - + ***Delete account:*** If a user no longer wants to have an account on the site, they can delete their account on the account settings page and all the data that is associated with the account.
 - + ***User game history:*** Users will be able to view their game history on their user dashboard.
- + ***Admin dashboard:*** Admins will be able to view their admin features (such as user account removal and password reset).
 - + ***Admin user account removal:*** An admin has the ability to permanently delete user accounts.
 - + ***Admin password reset:*** An admin has the ability to reset the password of a user account.

Deployment

I will deploy the Flask project with Heroku. After creating a Heroku account and installing the Heroku CLI, I will create a file named Procfile in the project’s root directory and update the requirements.txt file and commit the files to Git. Then, I will use the Heroku CLI to create a Heroku app and then push the Git repository to the remote branch created with the Heroku create command. Afterwards, the building and deployment process will occur; the app will be online at that point and will be able to be opened on the web browser.

Features That Will Be Accomplished During the Milestones

M1 (2/6 - 2/16): Finalize Design and Features + Database Setup

- + Wireframes
- + Database ERD
- + Initial database setup

M2 (2/16 - 2/29): User and Admin Account Management

- + Account creation
- + Login functionality
- + Admin login functionality
- + User authentication
- + Password hash/encryption
- + Logout functionality

- + User dashboard (general UI)
- + Admin dashboard (general UI)
- + Account settings
- + Delete account functionality
- + Admin user account removal features
- + Admin password reset feature

M3 (3/1 - 3/30): Game Implementation

- + Home screen
- + Game Implementation
- + Get hints mechanism
- + Submit solution mechanism
- + Get solution mechanism
- + Game end screen

M4 (3/30 - 4/10): Game Metrics Display for Users and Admin

- + User dashboard
 - + User game history
- + Admin dashboard
 - + User data display

M5 (4/11 - 4/16): Testing