

# Project Proposal

9조

# Contents

## 1 | Project Overview

- Concept
- How to use AI
- Implementation

## 2 | Expected Effect

## 3 | Development tools

## 4 | Plan and role

Section 1

# Project Overview

# Concept

Learning programming as **fun and easy** as a game

# Purpose



## Motivation

Reward for solving  
problems



## Making a habit

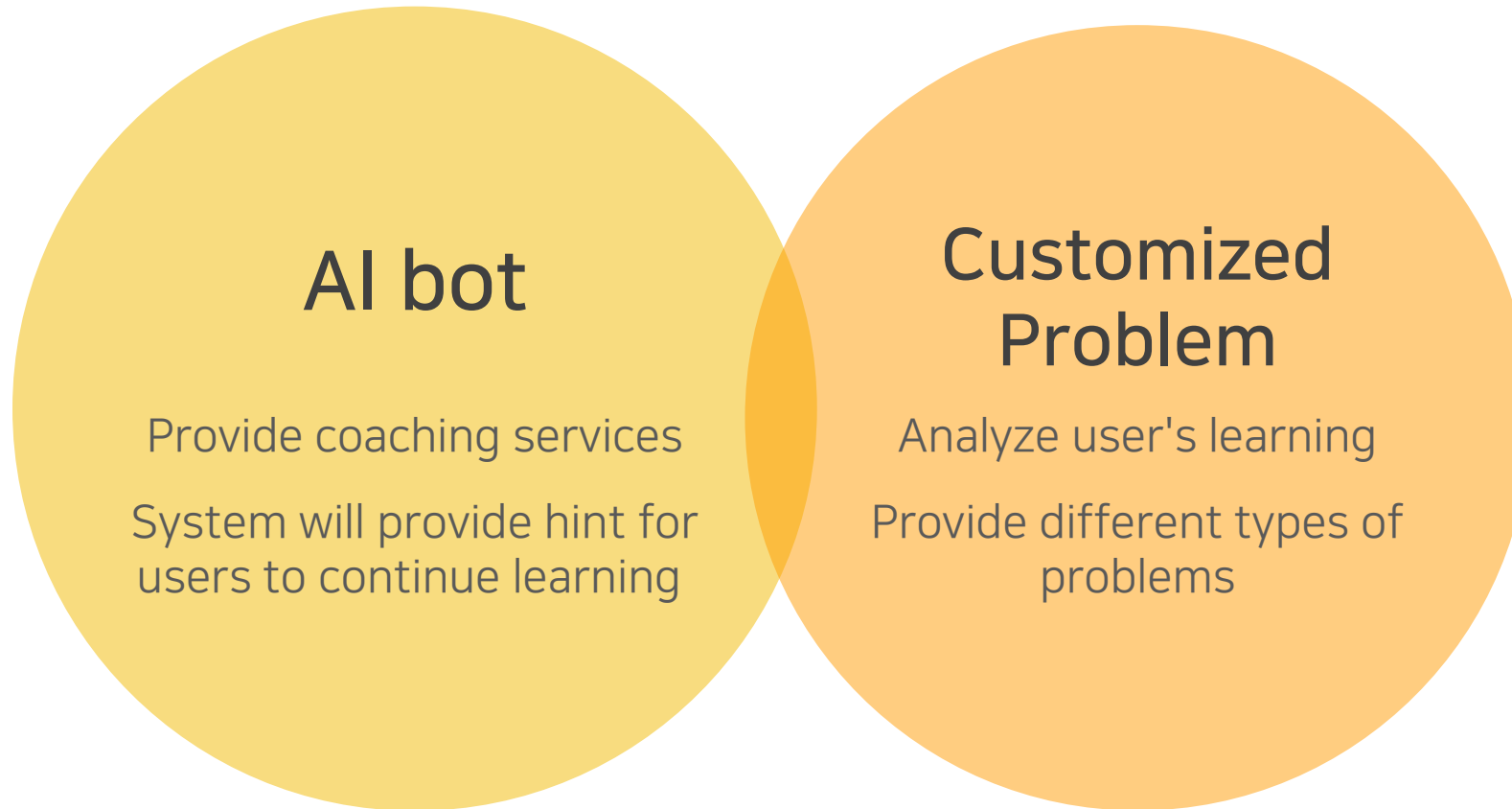
Create a habit to  
continue learning



## Positive feedback

Effective learning with  
hints and feedback

# How to use AI



# How to use AI

## Chat GPT

A sibling model to InstructGPT

Trained to **follow an instruction in a prompt and provide a detailed response**

## Codex

A model that powers  
GitHub Copilot

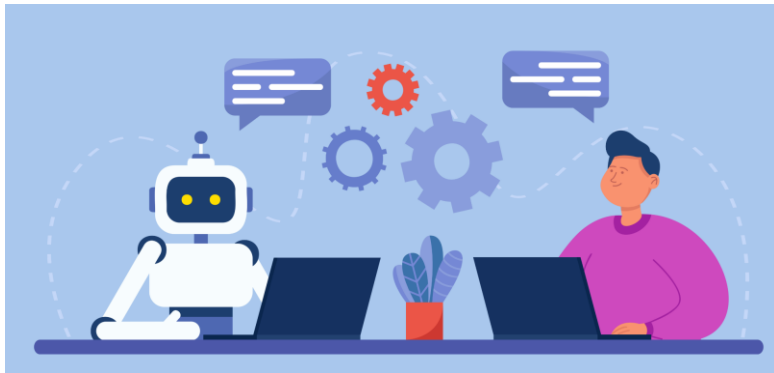
Codex can **now interpret simple commands in natural language and execute them on the user's behalf**

# Implementaion

**Problem Solving**

Feedback

Reward  
System



- **Practice mode**  
Solving problem and studying freely
- **Rank mode**  
Compete based on rank mode scores



# Implementaion

Problem  
Solving

**Feedback**

Reward  
System

- **Provide Hints**  
Help solve a problem
- **Q&A**  
Answers to questions asked by users
- **Review Note**  
Review the questions user got wrong



# Implementaion

Problem  
Solving

Feedback

**Reward System**

- **Attendance Reward**  
Solve problem every day
- **Scoring**  
Solve problem in given opportunity and time
- Changing the theme or get items with rewards



Section 2

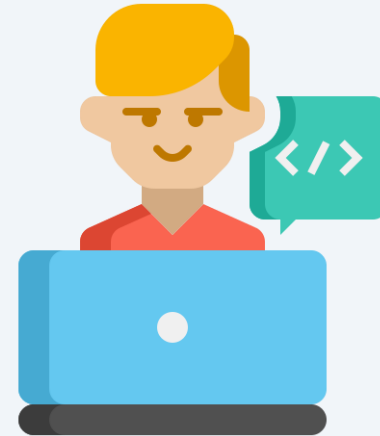
# Expected Effect

# Expected Effect



## Beginners

- Targeted on adult learners
- Learn not only the concept of coding but what can be useful on the field
- Keep motivated!



## Developers

- Learning new languages in a compelling, low-burden way
- Game-based approach

# Expected Effect

## Problem

**32. Longest Valid Parentheses**

Hard 10.7K 339

Companies

Given a string containing just the characters '(' and ')', return the length of the longest valid (well-formed) parentheses *substring*.



## Question about the Problem

One approach to solving this problem is to use a stack. We can iterate through the string, pushing the index of each opening bracket onto the stack. When we encounter a closing bracket, we pop the top index from the stack and use it to calculate the length of the current valid parentheses substring. We can repeat this process until we reach the end of the string, keeping track of the longest valid substring we encounter.

## AI Assistant

Code Review

Code Refactoring

User-customized hint



## Answer of the Question

Here are some additional hints that may help you solve the problem:

1. Initialize a stack and push -1 onto it. This will act as a starting point for the length of the first valid substring we encounter.
2. For each character in the string:
  - \* If the character is '(', push its index onto the stack.
  - \* If the character is ')':
    - \* Pop the top index from the stack. If the stack is empty after popping, push the current index onto the stack.
    - \* Calculate the length of the current valid substring using the difference between the current index and the top index on the stack.
    - \* Update the longest valid substring if necessary.
3. After iterating through the string, return the length of the longest valid substring.

I hope these hints help you solve the problem!

Section 3

# Development tools

# Development tools

Front-end



**React**  
Cross-platform  
application



**Photoshop**  
Image  
Component  
Design



**Language**  
HTML  
CSS



**MySQL**  
OpenSource  
DBMS



**Django**  
Python-based  
Web framework



**Language**  
Python

# Development tools

Project Management



Discussion



Version Control



Collaboration



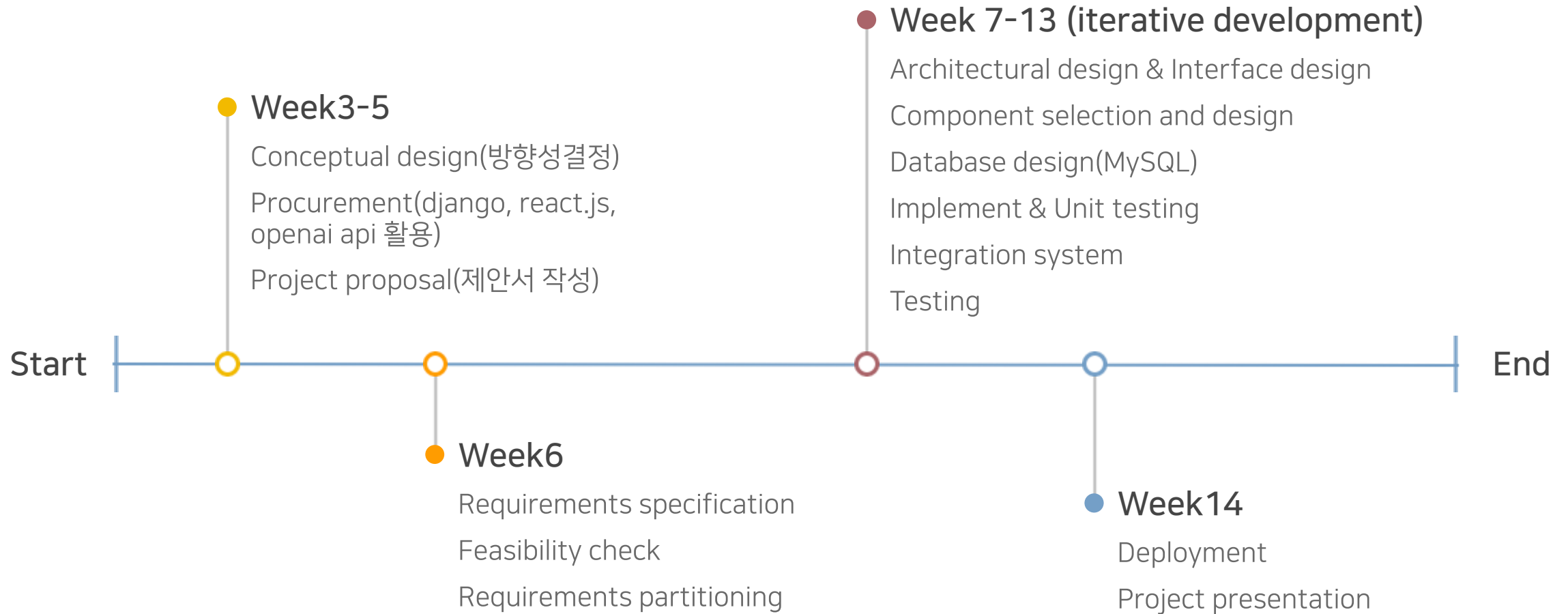
Section 4

# Plan and Role

# Plan

- 단기간 소규모 웹 프로젝트 → Component-based, Agile
- 요구명세서, 설계명세서 요구됨 → 초기 계획 수립
- 반복 진화 모델 → 잦은 delivery

# Plan



# Role



신새별

Front-end



이하은

Front-end



권혁준

Back-end



김민성

Back-end



박주봉

Back-end



최현진

Back-end

Section 5

# Q & A