Venus

Your personal game assistant

Members: Kathareeya Atthajaroon, Erick Bravo, Daniel Esquivel, Jazlyn Fernandez, Angelina Nantanapramoth, Juan Vazquez

Project Vision

- In-game machine learning AI that assists struggling players
- Distinct from starter NPC helpers
  - Ability to observe and analyze user's game play
  - Dynamic interaction
  - Independent from the game
  - Learning as the user does, it emulates a human playstyle
- Level of guidance modulated to:
  - Prevent “cheating”
  - Still serve a challenge for user to keep their interest
  - Create gameplay that scales with the user

Problem Statement

- Adding an AI helper, Venus, into a game where the user is having difficulty
  - Venus will analyze the user's playing-style and will adapt to the level of difficulty
  - After analysis, Venus will offer tips/hints to assist the user, but the tips won't be excessive
- Tutorials in games are often too complicated for the user to comprehend
- Venus will make the user's experience more fun and achievable on any level of difficulty
- End goal is to encourage users to play any kind of game despite the level of difficulty
  - User's experience will positively benefit with the inclusion of Venus

Requirements

01 An interactive open source game with a mostly well maintained code base

02 Design and build a game guide named “VENUS”

03 Gameplay shouldn’t be too easy or hard; smooth sailing with the game guide

04 Assistant without actively changing user input

Process

- A mod/addon/DLC to the existing game
- Anything the player passes thru will pass thru Venus then to the game
  - Player → Venus → Game
  - Game → Venus → player
- Run a model/predictive method that will try to get the best outcome for the user
  - Shown as hints via the sprite character
- Game input sent to Venus to learn from the outcomes
- Data saved in original game section, with a parallel section for Venus
  - Both are pulled simultaneously at start of game
- Timeline Draft

  In depth research on AI usage in game development → find source code for open source game → decide on best language to use → create user stories and mock ups → identify potential risks → develop

Risks

1. Being able to integrate with and modify an existing code base.
2. Finding suitable open source game.
3. Reinforcement of right strategy may take too long.

Plans

1. Find a FOSS game with a tested codebase
2. Allow AI to play the game itself, develop strategies, then compare to what player has done.

Features

- Learns from user’s play-style
- After many "deaths/losses/rage" quits, Venus can offer its principle role in offering advice/tips, *actual help*
- Can be put in a view only mode
    - Observes the player play and then offer advice at the end of a play session
- Also only have access to the players view
    - Still have a fog enabled so it can only experience it at the same time the player does

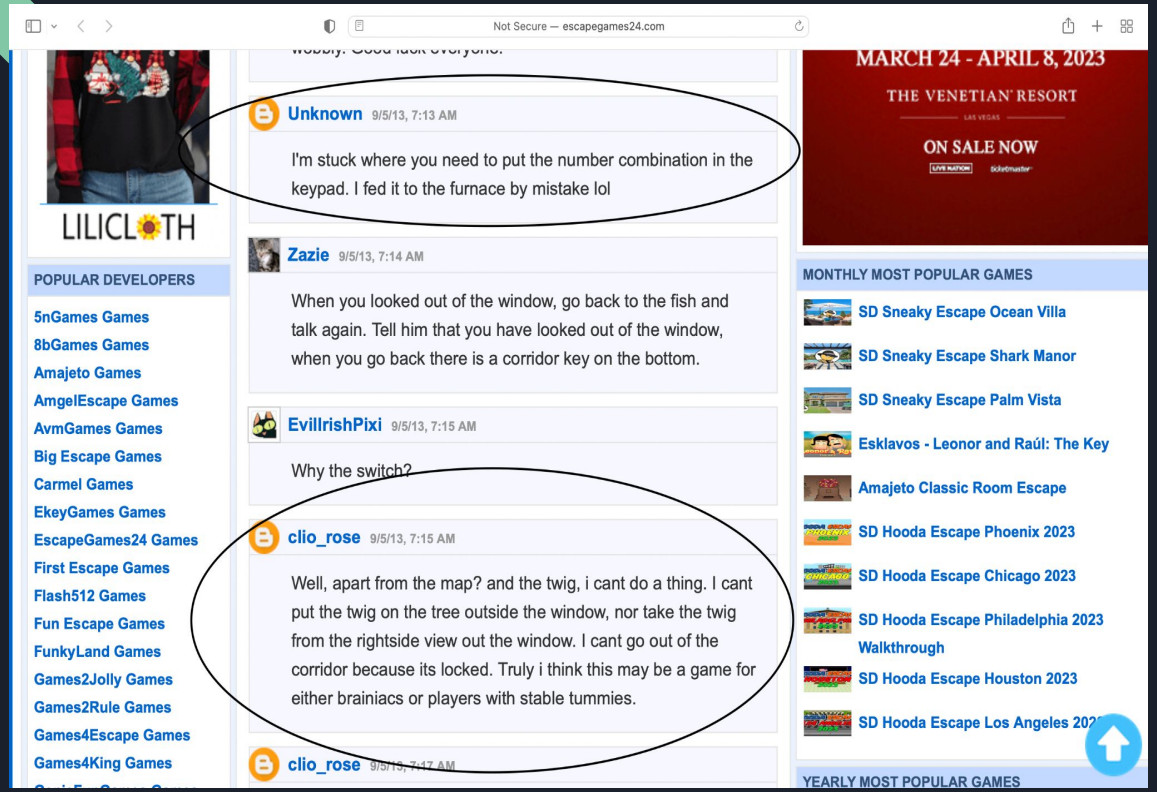
Possible Game Choice

An escape room kind of game

Issues encountered:

- Too little context
- Much time spent randomly clicking

The House Reviews



- Venus can address these issues in several ways
- Methods:
  - Give hints on when to use certain items
  - Provide a sense of direction if player gets stuck on a certain part for a period of time

Questions?