TODOs

- For each category create debate AI group which consistes of external model APIs
  - Each coder group shall contain of one coder and one regular model of same model family, for example qwen + qwen-coder
  - Expose the debate group as single "Model" / "Agent"

For later

- Enable Tool Calling for chosen Ollama Model(s) that will be used with Ollama Gemini
- TASK: 007 Hello Ollama Gemini [Gemini@Ollama]
- For each category create debate AI group
  - Each coder group shall contain of one coder and one regular model of same model family, for example qwen + qwen-coder
  - Expose the debate group via Ollama for API access
- Make sure that our scripts present its work like qwen code cli: https://github.com/QwenLM/qwen-code
- Patch Ollama Gemini with changes we want - visual, functional, etc
- Enable Goose worker
- Fixers: Support for Goose

In progress

- https://chutes.ai/app - Investigate the use
- https://opencode.ai/docs/providers - Opencode with Ollama models
- Define the `CONFIGURATION.sh` to override parameters from the global scope (`.bashrc`)
- Integrate RAG with Fixers to pickup from out memory database
- Fixers: Auto detection between AI CLIs sorted by priority - claude, goose, qwen, etc
- Web UI for tasks with APIs (instead of manual editing). After we create or edit task submit with confirmation sends it to execution to the AI
- Recipes for adding more tests (each test scenario is defined by the recipe)

Completed

- FIXME: Audio model for 7B does not exist on Ollama [Claude]

- Make sure that every model before it is installed is checked if exists in Ollama, as fallback pull it and apply from Hugging Face [Claude]

- TASK: 003 Testing Scripts [Claude]

- FIXME:

  ```
   ✘ python3 Scripts/AudioTemplates/text_to_speech.py 'Hello, this is a
  test'
    Traceback (most recent call last):
    File "Builder/Scripts/AudioTemplates/text_to_speech.py", line 10, in
  ```

```
<module>
      import soundfile as sf
  ModuleNotFoundError: No module named 'soundfile'
``` [Claude]
```

- Command for CLI AI has to be customizable with the def. val. and to run in headless mode

- Fixers: Add support for multiple `Fixers` - Besides `Claude` add `Qwen` as option/default one [Claude]

- Add all Upstreams supported: GitLab, GitFlic, Gitee, GitVerse

- TASK: 004 Cover all models by all supported Builder categories [Claude]

- Write log files during the testing [Claude]

- TASK: FIXME: 005 Make sure that tests are running with no errors or warnings [Qwen]

- TASK: 006 Hello Qwen [Qwen]

- Using local project-level Gemini fork with Ollama