

4.3 Snapshots of Working Prototype

```
== Starting Project Generation Pipeline ==
1. Parsing project requirements...
Calling language model...
Model language responded successfully
Detected: python | fastapi | REST API
Reasoning: User explicitly requested FastAPI, which is a Python framework for building REST APIs.
Calling language model...
Model language responded successfully
Error with API key 2: 'choices'
Calling language model...
Model language responded successfully
Error with API key 4: 'choices'
Calling language model...
Model language responded successfully
Generated project structure for python using fastapi
2. Decomposing tasks...
Calling reasoning model...
Model reasoning responded successfully
Error with API key 2: 'choices'
Calling reasoning model...
Model reasoning responded successfully
Raw decomposition response:
[
  {
    "name": "create_fastapi_app",
    "description": "Initialize FastAPI application and define the main entry point",
    "subtasks_necessary": false,
    "function_name": "app",
    "parameters": {},
    "return_type": "FastAPI",
    "file_path": "./fib_api/main.py",
    "language": "python",
    "framework": "fastapi",
    "implementation_details": {
      "TYPE": "application",
      "expected_loc": 15,
      "to_be_coded": true,
      "logic": "1. Import FastAPI from fastapi\n2. Creat..."
  }
]
cleaned response:
[
  {
    "name": "create_fastapi_app",
```

FIGURE 4: PROTOTYPE SNAPSHOT 1

In this step, the user enters the project details and runs CodeCodez. The system passes the prompt to a language model, which automatically detects the most suitable programming language and framework. It then generates a detailed description of the prompt in a structured way so that the reasoning model can clearly understand the project requirements. Finally, the reasoning model creates an iterative task tree that breaks down the project into smaller tasks.