0x22E

Oleg Sidorenkov
Maria Tsvetkova
Ksenia Poliakova
Anton Timonin
Egor Timonin

Project Description

Web editor for reproducible pipelines designed for creating, editing, validating specifications with a library of predefined and custom types that follow an ML experiment domain model defined as a semantic network represented in YAML. The editor could be used as an extension in VS Code or as Web App (WPA) served from website.

- **Store Specifications:** allows users save specification either by their own or via customizable autosave.
- **Edit Specifications:** built-in editor allows users to easily edit pipelines specifications without need to leave website or IDE.
- **Specifications Validation:** automatically checks job specifications for syntax errors, helps validate pipeline scripts, ensuring they are error-proof without need to execute.
- **Code Hints:** includes code highlighting, autocompletion and prompts (e. g. entity name automatic suggestions) and highly customizable linting.
- **Repository Integration:** all the changes can be applied to a GitLab/GitHub repository specification file.
- **Framework Integration:** specification state changes affect the MLOps platform via CI/CD framework integration.

Team: 0x22E

Project repo: https://github.com/2Delight/mline

Personas

Name: Maxim

Age: 26

Gender: man

Location: Moscow, Russian Federation

Education: Bachelor’s Degree, Higher School of Economics

Job Title: Backend Developer in Yandex, on-site

Income: 260,000₽-350,000₽

Needs: Automate release delivery and software quality checks. Store and manage different versions of pipeline specification.

Motivations: Wants to find the most convenient way to meet his needs.

Pain Points: Doesn’t want to manually run tests and linters every single time, feels stress while deploying new release manually because infrastructure becomes inconsistent

Persona's journey: Maxim has found a new VS Code extension that allows to execute pipelines, he liked the jobs reports visuals, therefore, decided to embed it in the company’s tech stack

Personas

Name: Nikita

Age: 20

Gender: man

Location: Novosibirsk, Russian Federation

Education: Bachelor’s Degree, Novosibirsk State University

Job Title: Project Manager in Sber, hybrid

Income: 130,000₽-200,000₽

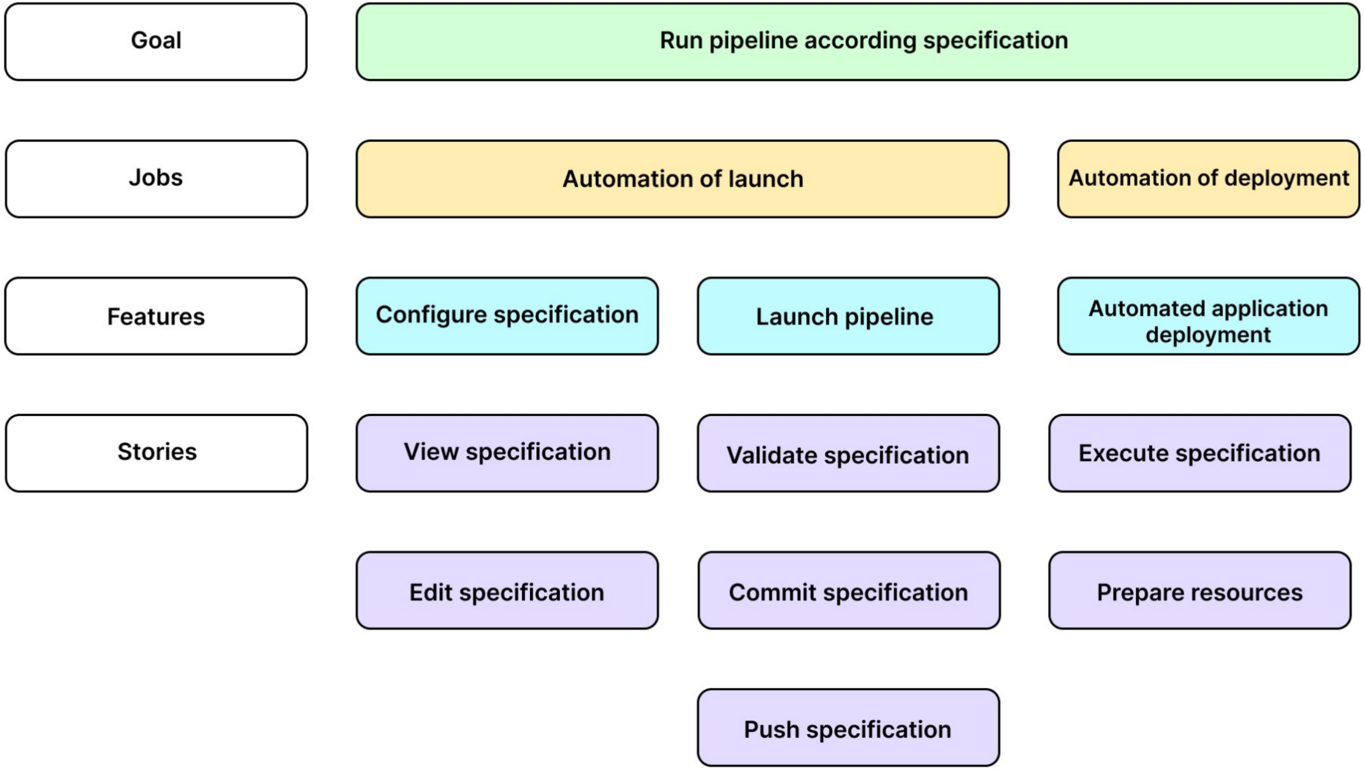
Needs: Run tasks that are valuable for business

Motivations: Wants a simple UI to interact with application state

Pain Points: He doesn’t want to bother developers in order to manipulate software state, for instance, deliver releases or rollback unsuccessful ones. Also, he doesn’t want to get various metrics, such as TTM, test coverage and code quality from different places, he wants to see them structured in one place to save time.

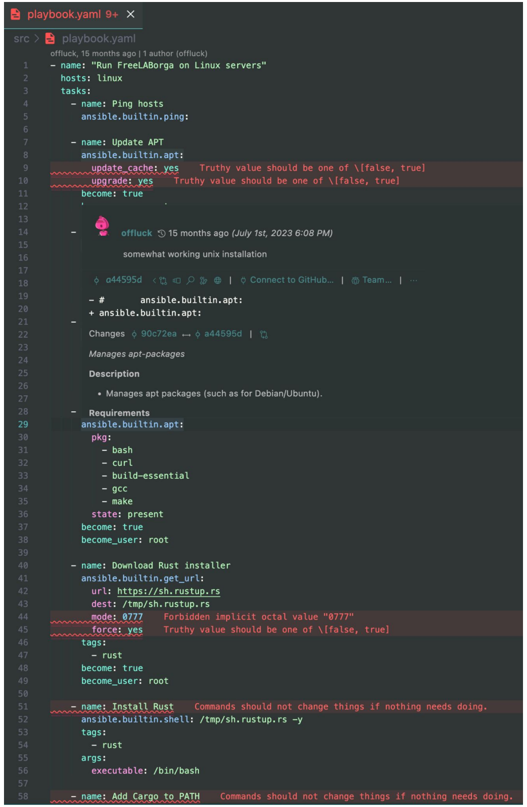
Persona's journey: He asked developers if there is a way for him to control and modify product state in exceptional situations

Story Map



Goal	Run pipeline according specification		
Jobs	Automation of launch		Automation of deployment
Features	Configure specification	Launch pipeline	Automated application deployment
Stories	View specification	Validate specification	Execute specification
	Edit specification	Commit specification	Prepare resources
		Push specification	

Prototype User Interfaces (Wireframes)



offluck, 15 months ago | 1 author (offluck)

```
- name: "Run FreeLABorga on Linux servers"
  hosts: linux
  tasks:
    - name: Ping hosts
      ansible.builtin.ping:

    - name: Update APT
      ansible.builtin.apt:
        update_cache: yes    Truthy value should be one of \[false, true]
        upgrade: yes    Truthy value should be one of \[false, true]
      become: true
```

offluck 15 months ago *(July 1st, 2023 6:08 PM)*

somewhat working unix installation

a44595d | Connect to GitHub... | Team... | ···

```
- #       ansible.builtin.apt:
+ ansible.builtin.apt:
```

Changes 90c72ea ⟷ a44595d |

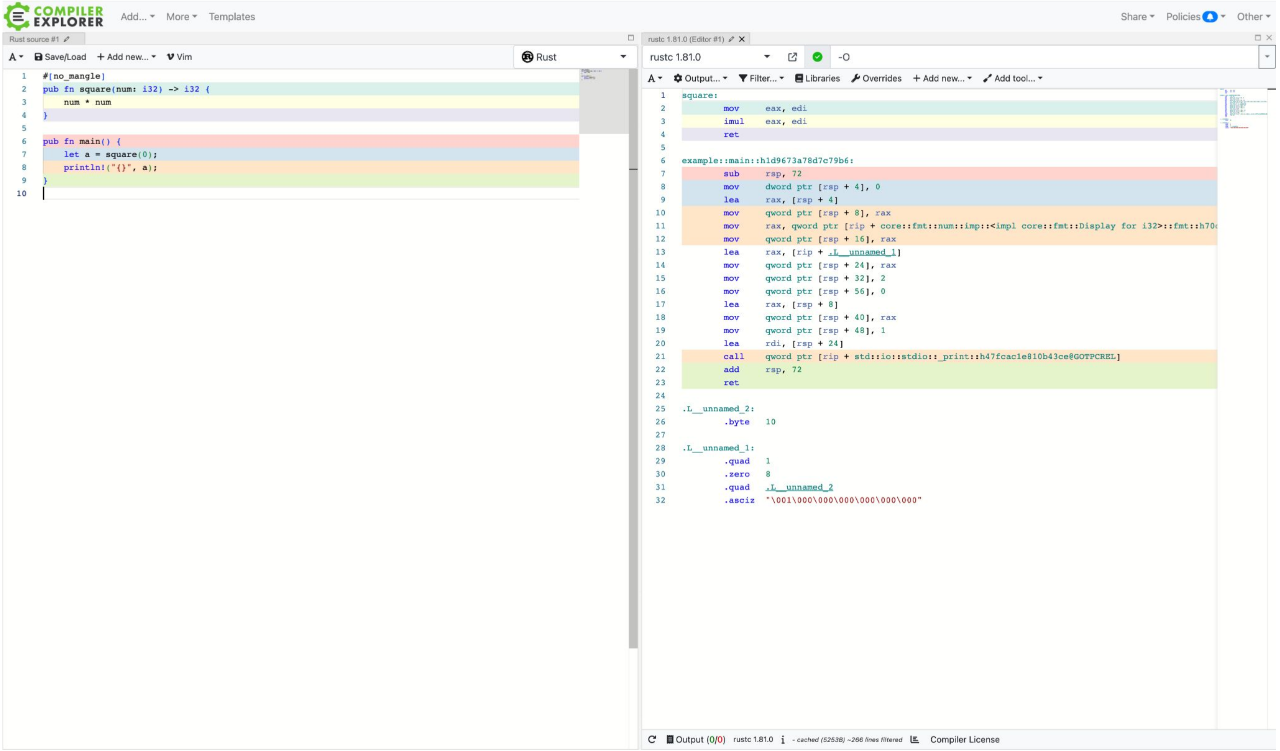
Manages apt-packages

Description

- Manages apt packages (such as for Debian/Ubuntu).

Requirements

```
      ansible.builtin.apt:
        pkg:
          - bash
          - curl
          - build-essential
          - gcc
          - make
        state: present
      become: true
      become_user: root

    - name: Download Rust installer
      ansible.builtin.get_url:
        url: https://sh.rustup.rs
        dest: /tmp/sh.rustup.rs
        mode: 0777    Forbidden implicit octal value "0777"
        force: yes    Truthy value should be one of \[false, true]
      tags:
        - rust
      become: true
      become_user: root

    - name: Install Rust    Commands should not change things if nothing needs doing.
      ansible.builtin.shell: /tmp/sh.rustup.rs -y
      tags:
        - rust
      args:
        executable: /bin/bash

    - name: Add Cargo to PATH    Commands should not change things if nothing needs doing.
```

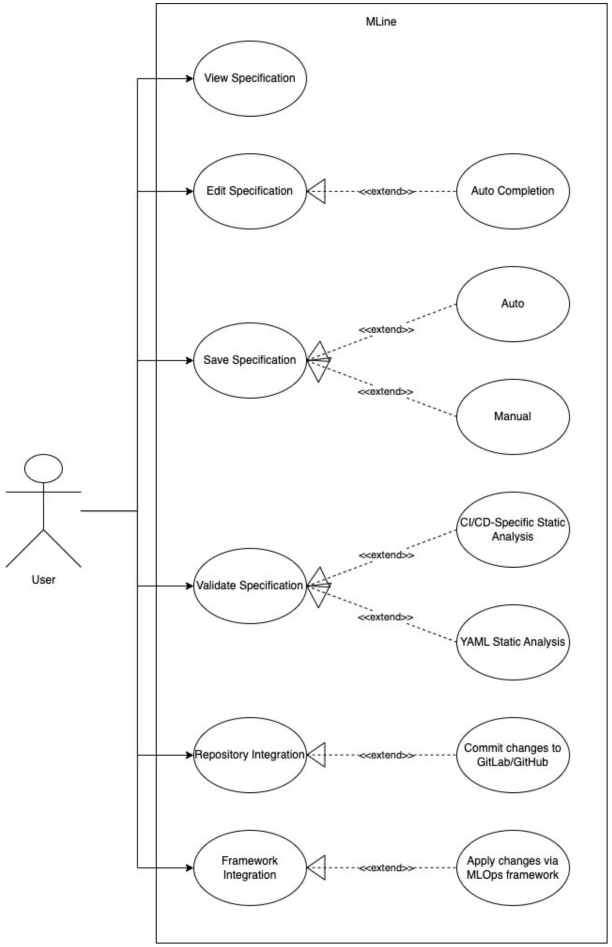


```
#[no_mangle]
pub fn square(num: i32) -> i32 {
    num * num
}

pub fn main() {
    let a = square(0);
    println!("{}", a);
}
```

```
square:
        mov     eax, edi
        imul    eax, edi
        ret

example::main::h1d9673a78d7c79b6:
        sub     rsp, 72
        mov     dword ptr [rsp + 4], 0
        lea     rax, [rsp + 4]
        mov     qword ptr [rsp + 8], rax
        mov     rax, qword ptr [rip + core::fmt::num::imp::<impl core::fmt::Display for i32>::fmt::h70c
        mov     qword ptr [rsp + 16], rax
        lea     rax, [rip + .L__unnamed_1]
        mov     qword ptr [rsp + 24], rax
        mov     qword ptr [rsp + 32], 2
        mov     qword ptr [rsp + 56], 0
        lea     rax, [rsp + 8]
        mov     qword ptr [rsp + 40], rax
        mov     qword ptr [rsp + 48], 1
        lea     rdi, [rsp + 24]
        call    qword ptr [rip + std::io::stdio::_print::h47fcac1e810b43ce@GOTPCREL]
        add     rsp, 72
        ret

.L__unnamed_2:
        .byte   10

.L__unnamed_1:
        .quad   1
        .zero   8
        .quad   .L__unnamed_2
        .asciz  "\001\000\000\000\000\000\000"
```

Output (0/0) rustc 1.81.0 - cached (52538) ~266 lines filtered Compiler License

Use Case Diagram



Interaction Analysis

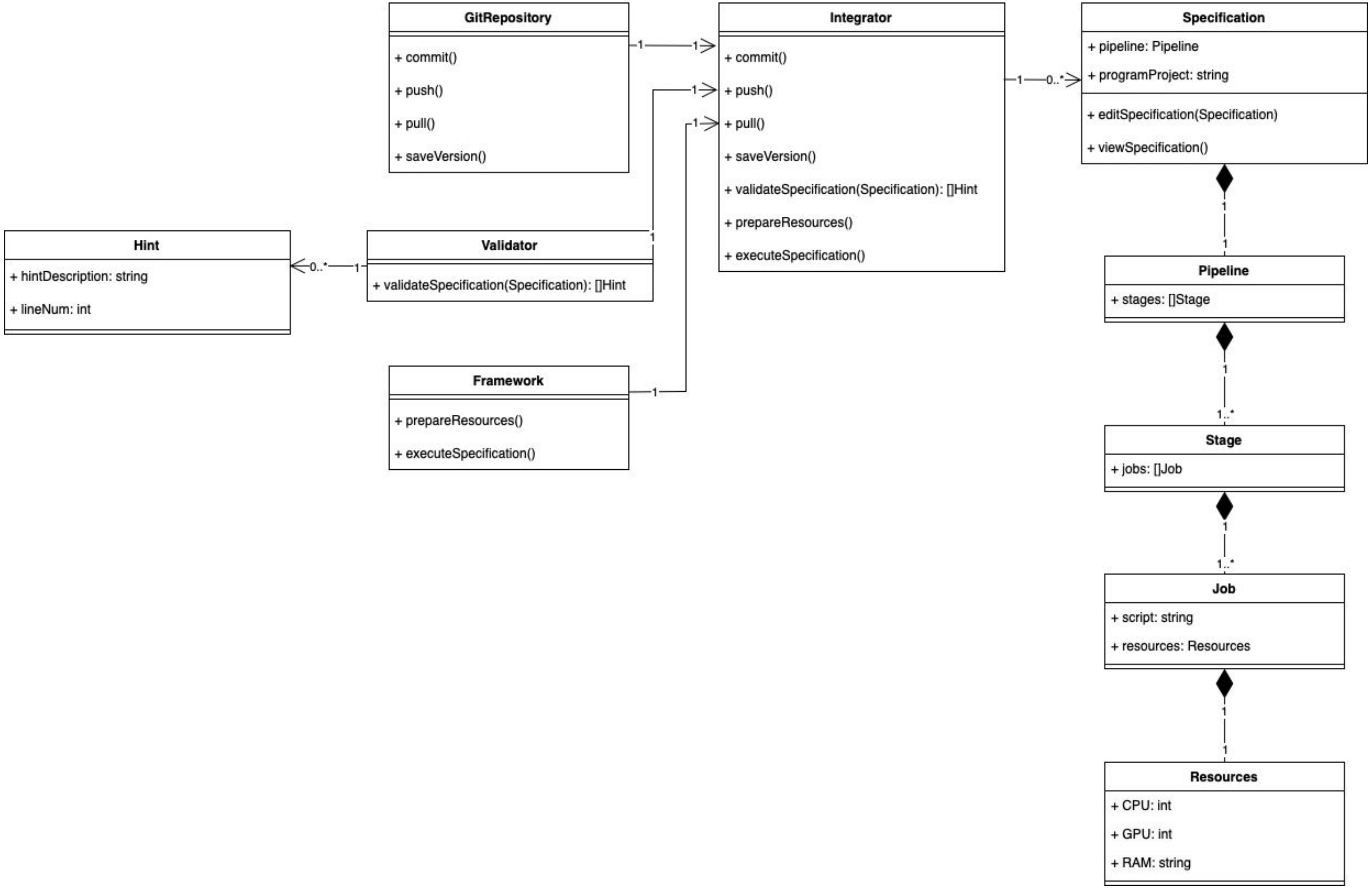
Cooperations table

Use case	Cooperation name	Used roles	Candidate classes
View specification	view	User views specification for particular repository.	Specification, GitRepository, User
Edit specification	edit	User edits specification for particular repository.	Specification, GitRepository, User
Validate specification	validate	Validator checks specification for mistakes.	Validator, Specification
Commit specification	commit	User commits specification to particular repository.	Specification, GitRepository, User
Push specification	push	User pushes specification to particular repository.	Specification, GitRepository, User

Interaction Analysis

Cooperations table			
Use case	**Cooperation name**	**Used roles**	**Candidate classes**
Execute specification	execute	The framework tracks changes in a remote repository and if there're execute specification that consists of a pipeline, which includes stages that contain jobs.	Specification, GitRepository, Framework, Pipeline, Stage, Job
Prepare resources	prepare	The framework gets data on the amount of resources needed and prepares them.	Specification, Framework, Job, Resources

Final Class Diagram



GitRepository

+ commit()
+ push()
+ pull()
+ saveVersion()

Integrator

+ commit()
+ push()
+ pull()
+ saveVersion()
+ validateSpecification(Specification): []Hint
+ prepareResources()
+ executeSpecification()

Specification

+ pipeline: Pipeline
+ programProject: string
+ editSpecification(Specification)
+ viewSpecification()

Hint

+ hintDescription: string
+ lineNum: int

Validator

+ validateSpecification(Specification): []Hint

Framework

+ prepareResources()
+ executeSpecification()

Pipeline

+ stages: []Stage

Stage

+ jobs: []Job

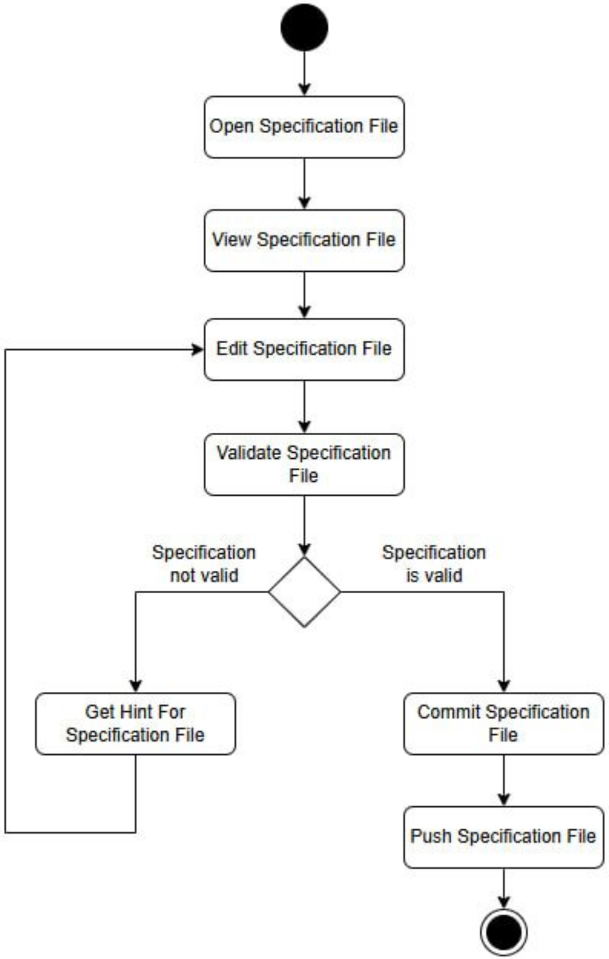
Job

+ script: string
+ resources: Resources

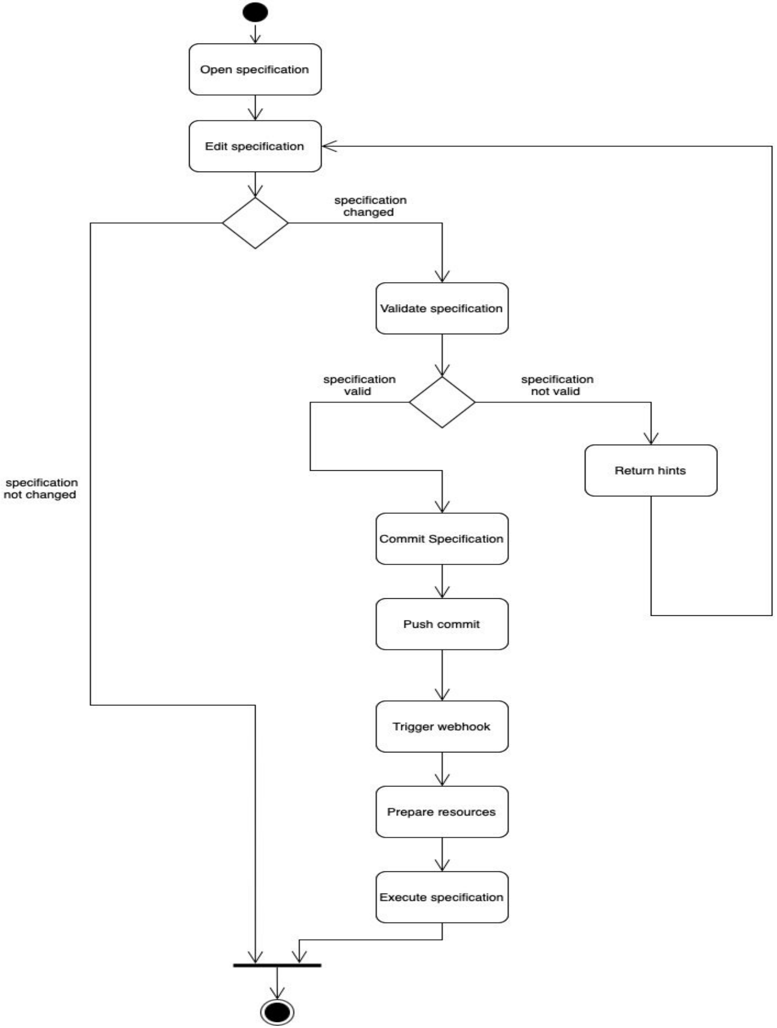
Resources

+ CPU: int
+ GPU: int
+ RAM: string

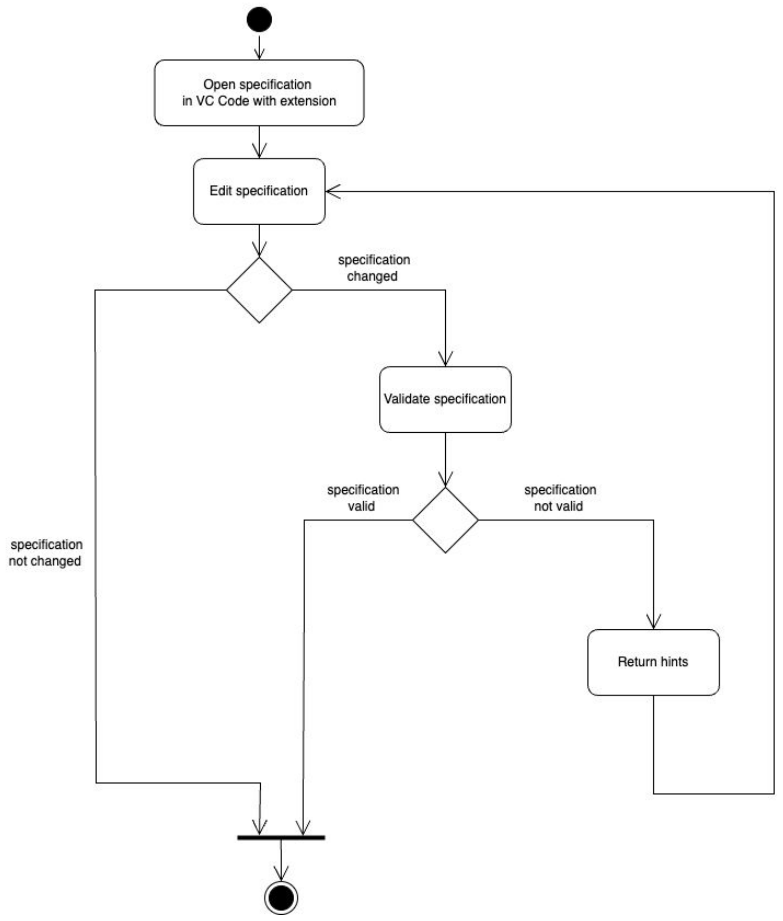
Detailed Behavior



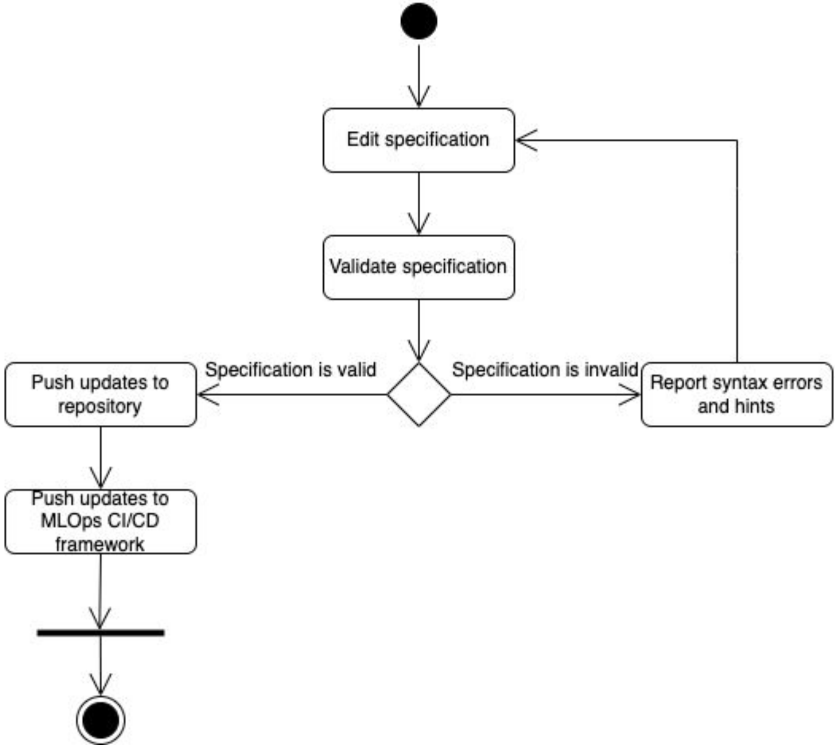
Detailed Behavior (etimonin)



Detailed Behavior (atimonin)



Detailed Behavior (osidorenkov)



Edit specification

Validate specification

Specification is valid

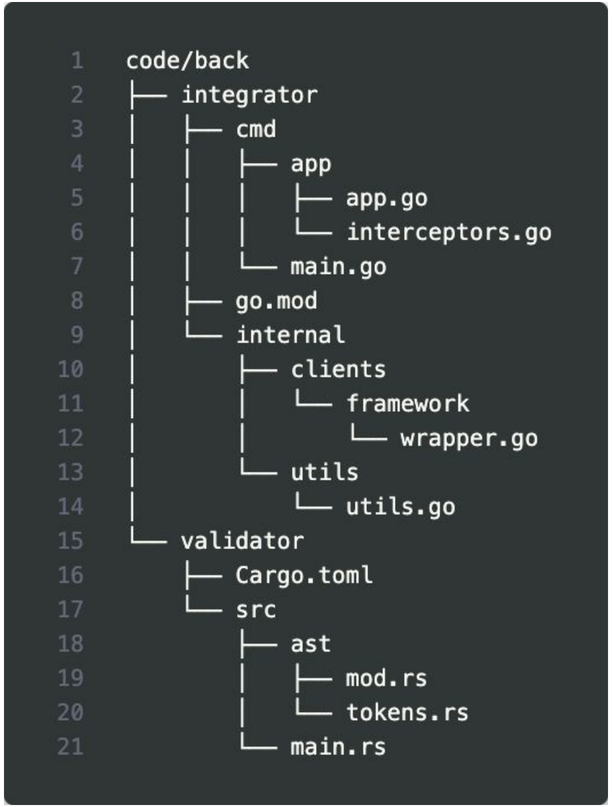
Specification is invalid

Push updates to repository

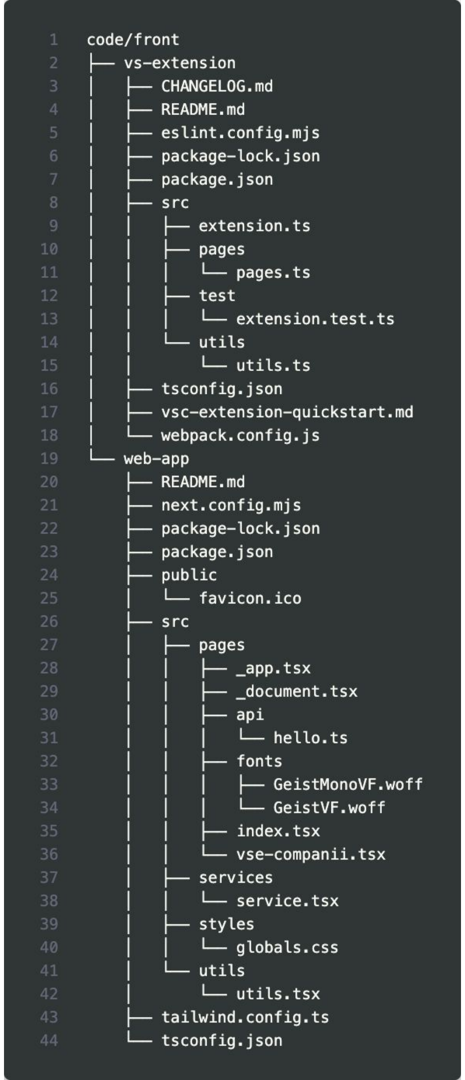
Report syntax errors and hints

Push updates to MLOps CI/CD framework

Repository Structure



```
code/back
├── integrator
│   ├── cmd
│   │   ├── app
│   │   │   ├── app.go
│   │   │   └── interceptors.go
│   │   └── main.go
│   ├── go.mod
│   └── internal
│       ├── clients
│       │   └── framework
│       │       └── wrapper.go
│       └── utils
│           └── utils.go
└── validator
    ├── Cargo.toml
    └── src
        ├── ast
        │   ├── mod.rs
        │   └── tokens.rs
        └── main.rs
```



```
code/front
├── vs-extension
│   ├── CHANGELOG.md
│   ├── README.md
│   ├── eslint.config.mjs
│   ├── package-lock.json
│   ├── package.json
│   ├── src
│   │   ├── extension.ts
│   │   ├── pages
│   │   │   └── pages.ts
│   │   ├── test
│   │   │   └── extension.test.ts
│   │   └── utils
│   │       └── utils.ts
│   ├── tsconfig.json
│   ├── vsc-extension-quickstart.md
│   └── webpack.config.js
└── web-app
    ├── README.md
    ├── next.config.mjs
    ├── package-lock.json
    ├── package.json
    ├── public
    │   └── favicon.ico
    ├── src
    │   ├── pages
    │   │   ├── _app.tsx
    │   │   ├── _document.tsx
    │   │   ├── api
    │   │   │   └── hello.ts
    │   │   ├── fonts
    │   │   │   ├── GeistMonoVF.woff
    │   │   │   └── GeistVF.woff
    │   │   ├── index.tsx
    │   │   └── vse-companii.tsx
    │   ├── services
    │   │   └── service.tsx
    │   ├── styles
    │   │   └── globals.css
    │   └── utils
    │       └── utils.tsx
    ├── tailwind.config.ts
    └── tsconfig.json
```

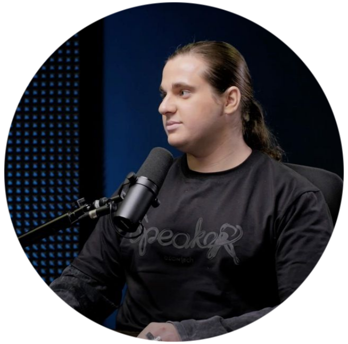
Team and Roles

Egor Timonin

Backend developer

Tsvetkova Maria

Backend developer



Oleg Sidorenkov
Infrastructure engineer
Teamlead - tg - @olegdayo

Ksenia Poliakova
Project manager,
QA Specialist

Anton Timonin

Frontend developer

Thank you for your attention!