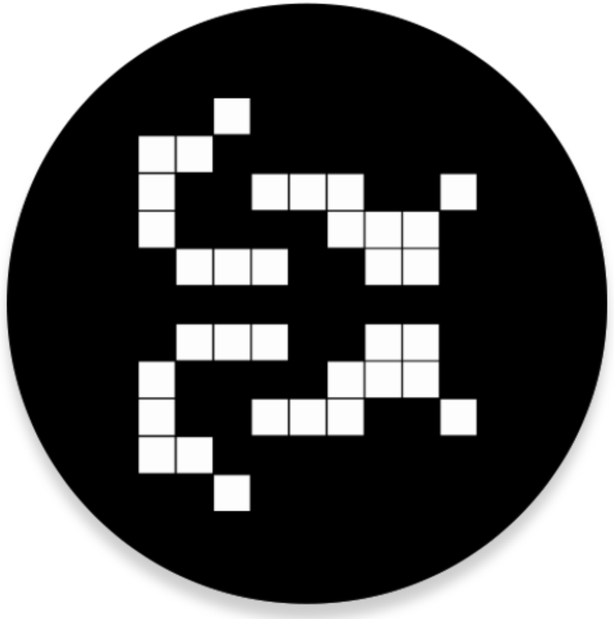


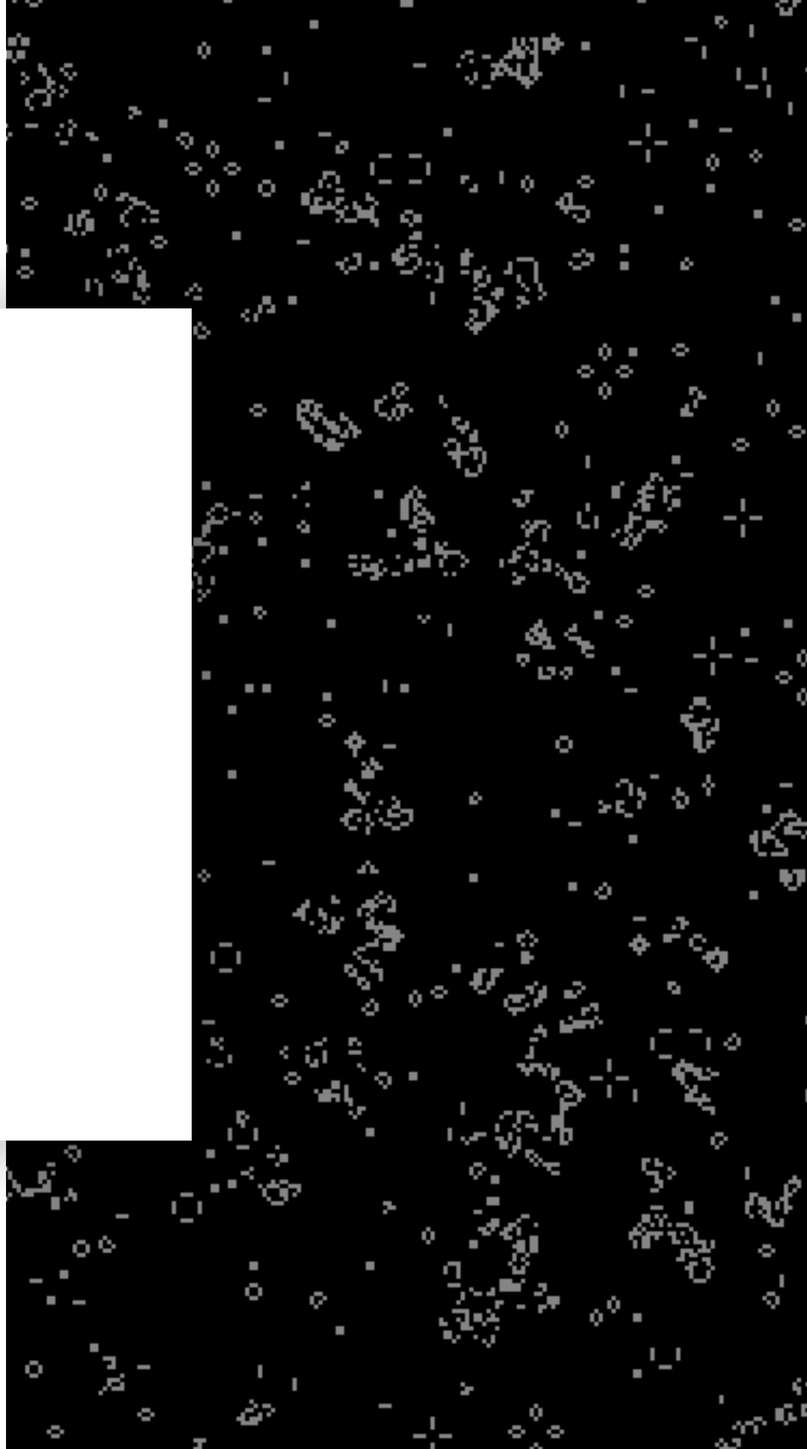
Better Life Game

Life Game 기능 확장 및 설계 개선

Design Patterns 2022 텀프로젝트 | 11 | 김혜성, 박서현, 박지수

Contents

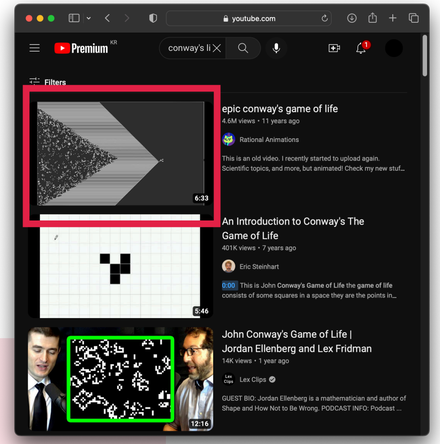
1. Project Abstraction
2. Project Design
3. Implementation
4. Results



Project Abstract

■ 기능 확장

예술적 요소 즐길 수 있도록 편의 기능 추가



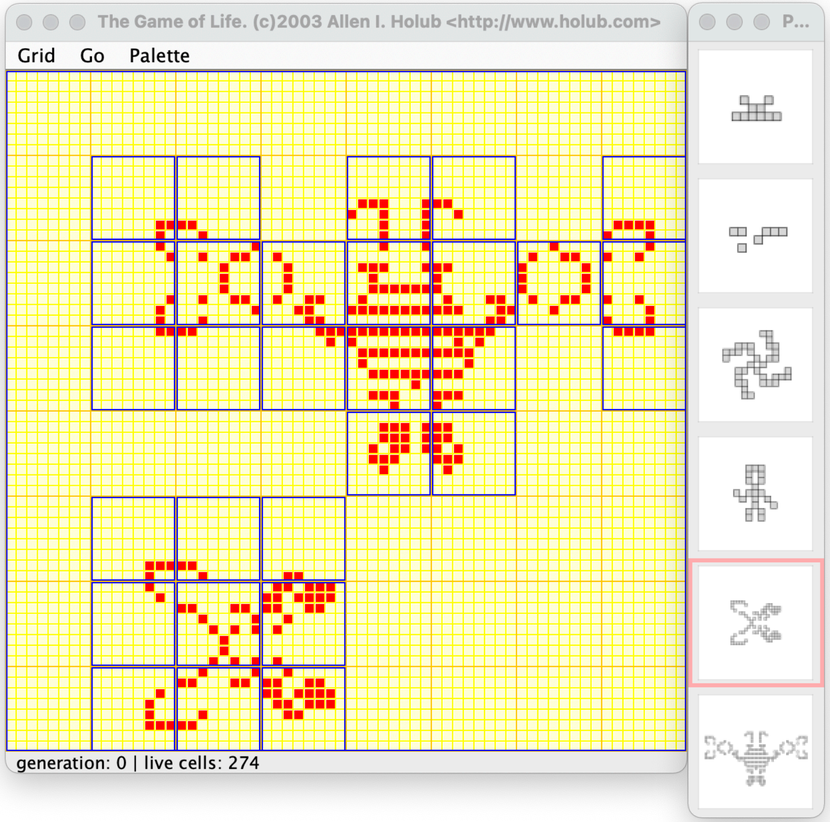
작은 크기의 cell 하나씩 누르며 패턴 그리기 번거로움

규모 큰 패턴 만들기 어려움

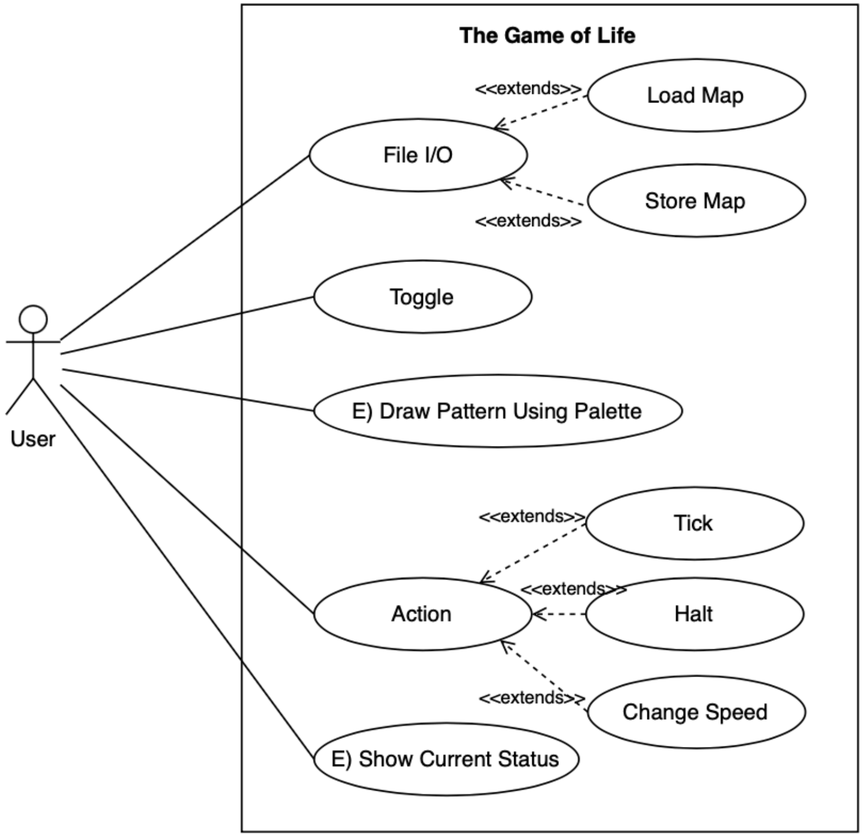
동시다발적 변화의 진행 상황 따라가기 어려움

패턴 가져다 붙일 수 있는 Palette 기능

Generation, Live cells 수 확인 가능한 status bar



Project Abstract



■ 설계 개선

Singleton, MVC, Observer, Strategy, Command Pattern 활용

1) SRP 위반 : View에서 모든 로직 수행

Model-View-Controller 책임 분리하여 **SRP 지킴**

2) OCP 위반 : 확장 어려움

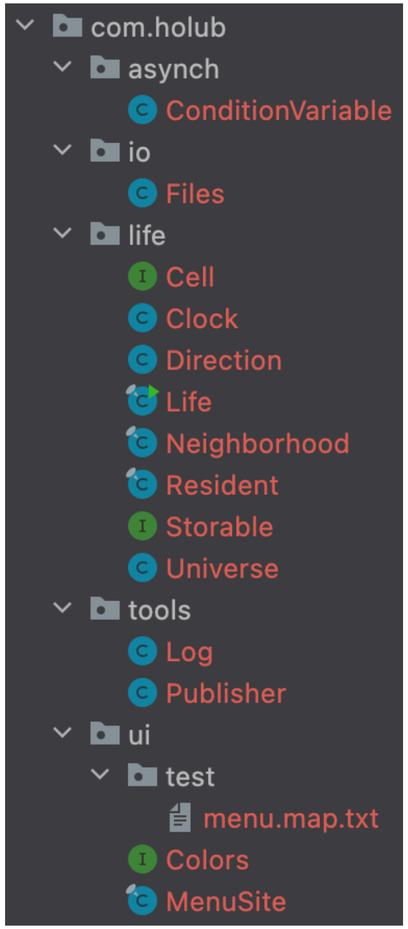
Observer(View)-Subject(Model) 패턴으로 **OCP 지킴**

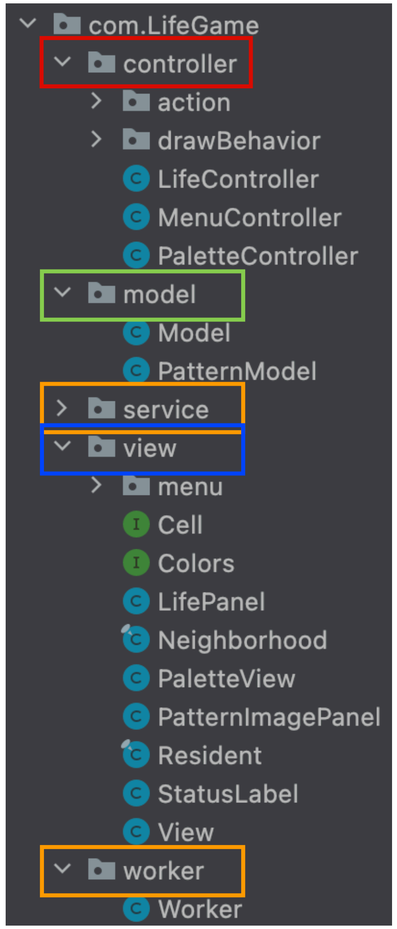
3) DIP 위반 : 의존 관계 많음

UI컴포넌트, Controller **Singleton** 구성하여 **DIP 지킴**

+ Palette 기능 추가 위해 **Strategy** 패턴, Menu OCP 위해 **Command** 패턴

Project Abstract





■ 설계 개선

Singleton, MVC, Observer, Strategy, Command Pattern 활용

1) SRP 위반 : View에서 모든 로직 수행

Model-View-Controller 책임 분리하여 **SRP 지킴**

2) OCP 위반 : 확장 어려움

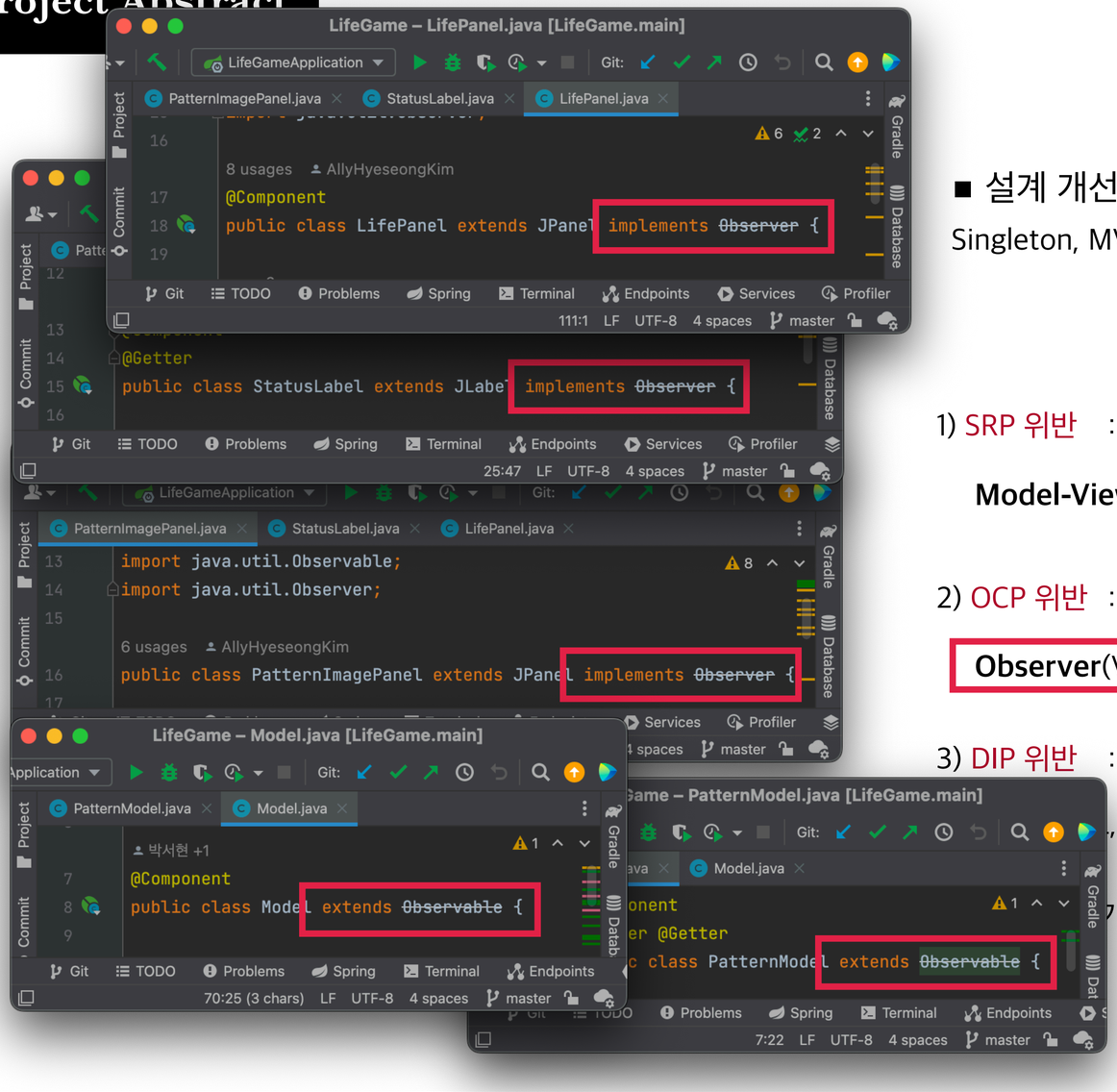
Observer(View)-Subject(Model) 패턴으로 **OCP 지킴**

3) DIP 위반 : 의존 관계 많음

UI컴포넌트, Controller **Singleton** 구성하여 **DIP 지킴**

+ Palette 기능 추가 위해 **Strategy** 패턴, Menu OCP 위해 **Command** 패턴

Project Abstract



■ 설계 개선

Singleton, MVC, Observer, Strategy, Command Pattern 활용

1) SRP 위반 : View에서 모든 로직 수행

Model-View-Controller 책임 분리하여 **SRP 지킴**

2) OCP 위반 : 확장 어려움

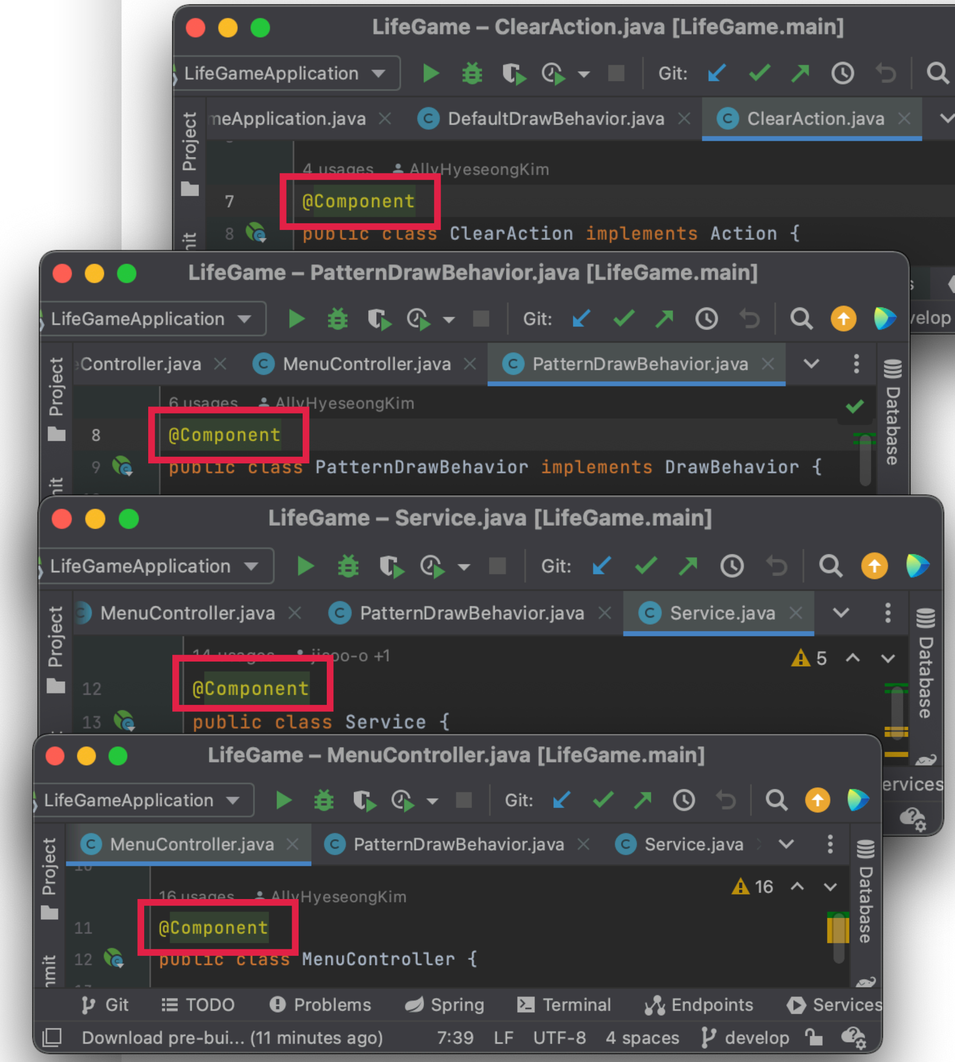
Observer(View)-Subject(Model) 패턴으로 **OCP 지킴**

3) DIP 위반 : 의존 관계 많음

, Controller **Singleton** 구성하여 **DIP 지킴**

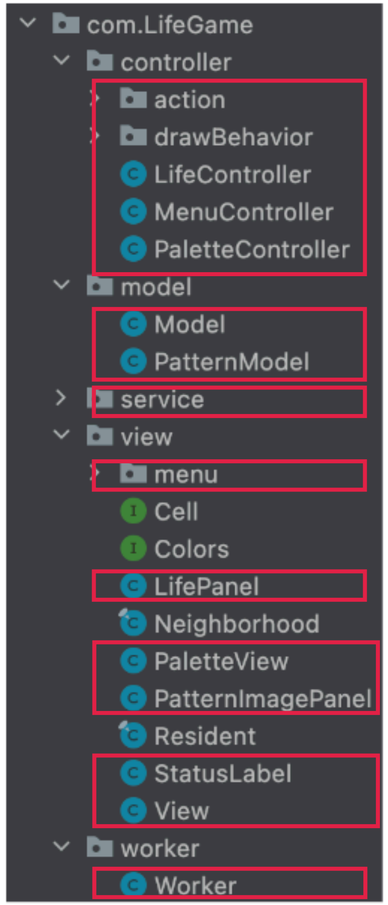
기능 추가 위해 **Strategy** 패턴, Menu OCP 위해 **Command** 패턴

Project Abstract



■ 설계 개선

Singleton, MVC, Observer, Strategy, Command Pattern 활용



1) SRP 위반 : View에서 모든 로직 수행

Model-View-Controller 책임 분리하여 **SRP 지킴**

2) OCP 위반 : 확장 어려움

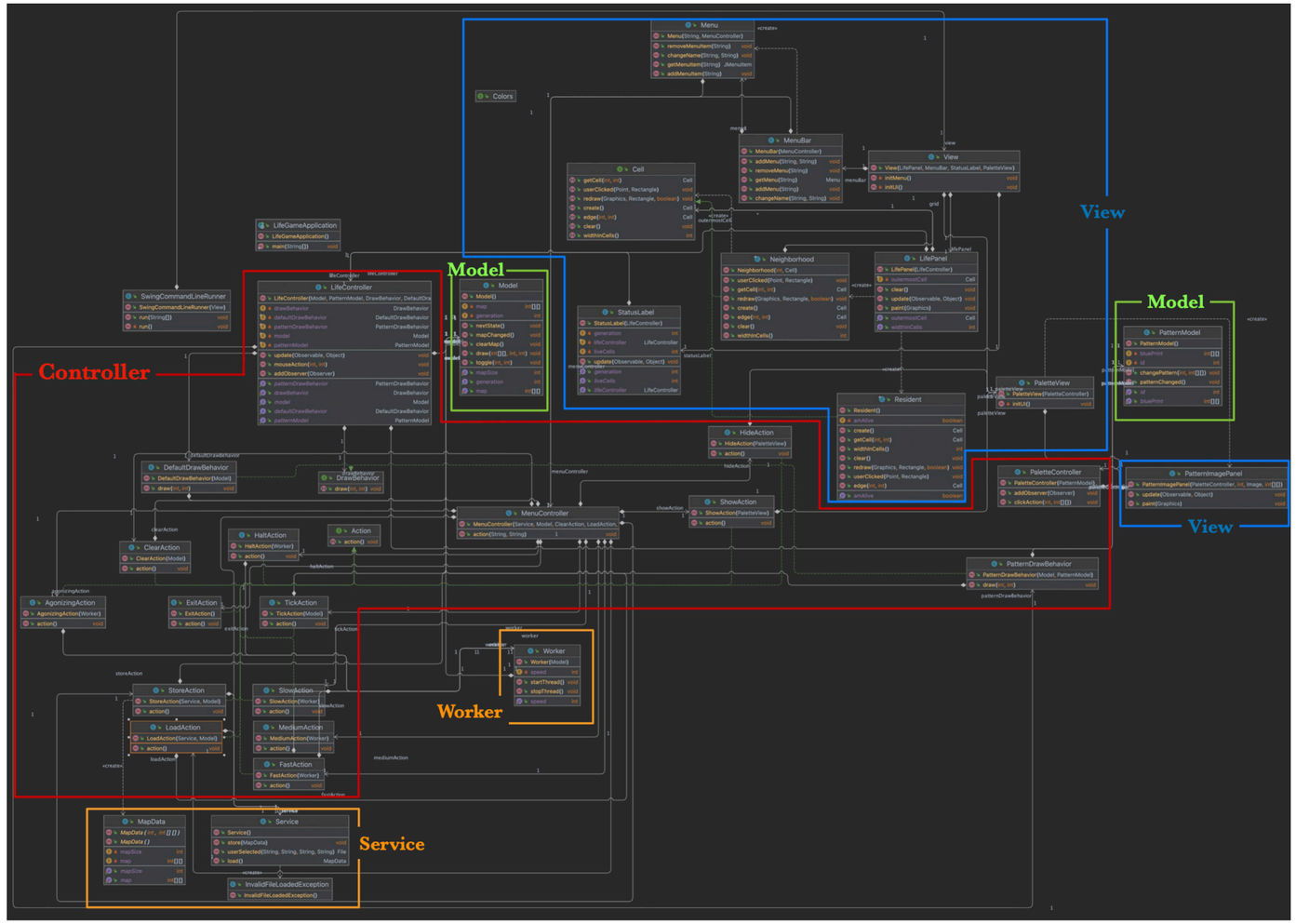
Observer(View)-Subject(Model) 패턴으로 **OCP 지킴**

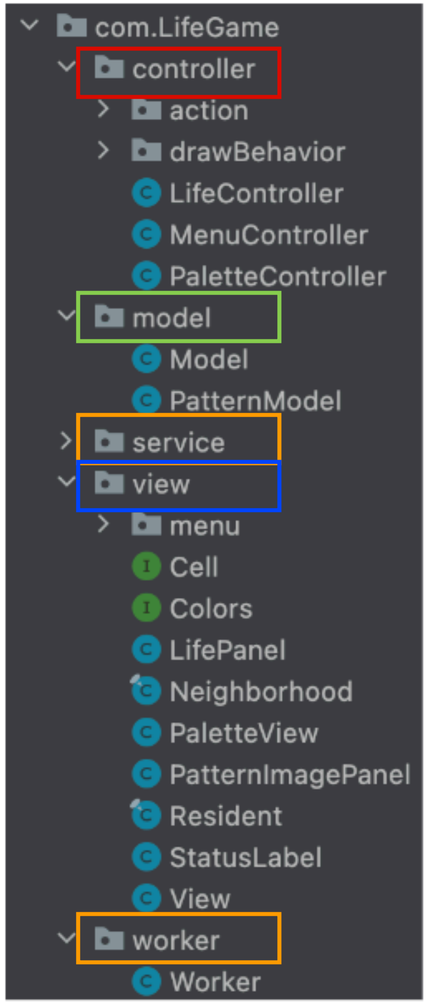
3) DIP 위반 : 의존 관계 많음

UI컴포넌트, Controller **Singleton** 구성하여 **DIP 지킴**

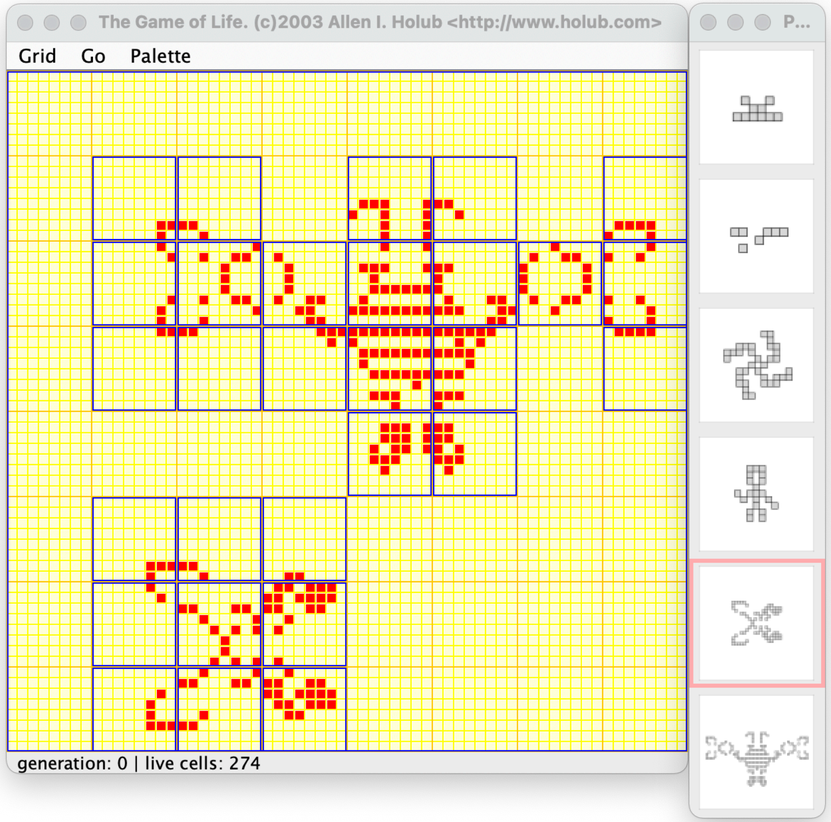
+ Palette 기능 추가 위해 **Strategy** 패턴, Menu OCP 위해 **Command** 패턴

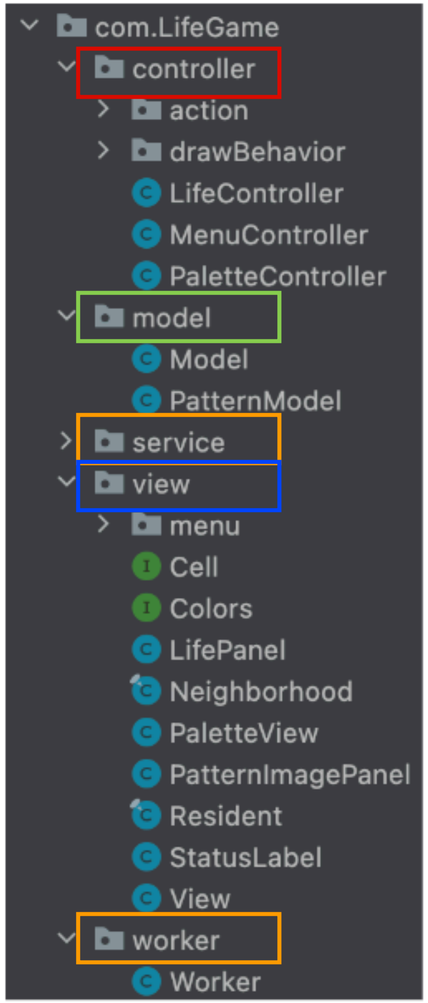
Project Design





Project Design





Project Design

LifeGame – StatusLabel.java [LifeGame.main]

```java
@Override
public void update(Observable o, Object arg) {
    if (o instanceof Model) {
        this.generation = ((Model) o).getGeneration();
        this.liveCells = 0;
        int[][] map = ((Model) o).getMap();
        for (int i = 0; i < ((Model) o).getMapSize(); i++) {
            for (int j = 0; j < ((Model) o).getMapSize(); j++) {
                this.liveCells += map[i][j];
            }
        }
        this.setText("  generation: " + this.generation + " | live cells: " + this.liveCells);
    }
}
```

LifeGame – PatternImagePanel.java [LifeGame.main]

```java
@Override
public void update(Observable o, Object arg) {
    if (o instanceof PatternModel) {
        if (((PatternModel) o).getId() == this.id) {
            this.setBorder(BorderFactory.createLineBorder(Color.PINK, thickness: 3));
        } else {
            this.setBorder(null);
        }
    }
}
```

```
com.LifeGame
  controller
    action
    drawBehavior
    LifeController
    MenuController
    PaletteController
  model
    Model
    PatternModel
  service
  view
    menu
    Cell
    Colors
    LifePanel
    Neighborhood
    PaletteView
    PatternImagePanel
    Resident
    StatusLabel
    View
  worker
    Worker
```

Project Design

LifeGame – Menu.java [LifeGame.main]

com > LifeGame > view > menu > Menu > Menu

LifePanel.java | PatternImagePanel.java | Menu.java

6 usages AllyHyeseongKim

```java
public void addMenuItem(String name) {
    JMenuItem menuItem = new JMenuItem(name);
    AllyHyeseongKim
    menuItem.addActionListener(new ActionListener() {
        AllyHyeseongKim
        @Override
        public void actionPerformed(ActionEvent e) {
            JMenuItem menuItem = (JMenuItem) e.getSource();
            JPopupMenu popupMenu = (JPopupMenu) menuItem.getParent();

            menuController.action(((JMenu) popupMenu.getInvoker()).getActionCommand(), e.getActionCommand());
        }
    });
    this.ac
    this.me
}
```

AllyHyeseongKim

```java
public void action(String menu, String menuItem) {
    this.actions.get(menu).get(menuItem).action();
}
}
```

void action()

com.LifeGame.controller.action.Action

LifeGame.main

- com.LifeGame
  - controller
    - action
    - drawBehavior
    - LifeController
    - MenuController
    - PaletteController
  - model
    - Model
    - PatternModel
  - service
  - view
    - menu
    - Cell
    - Colors
    - LifePanel
    - Neighborhood
    - PaletteView
    - PatternImagePanel
    - Resident
    - StatusLabel
    - View
  - worker
    - Worker

Project Design

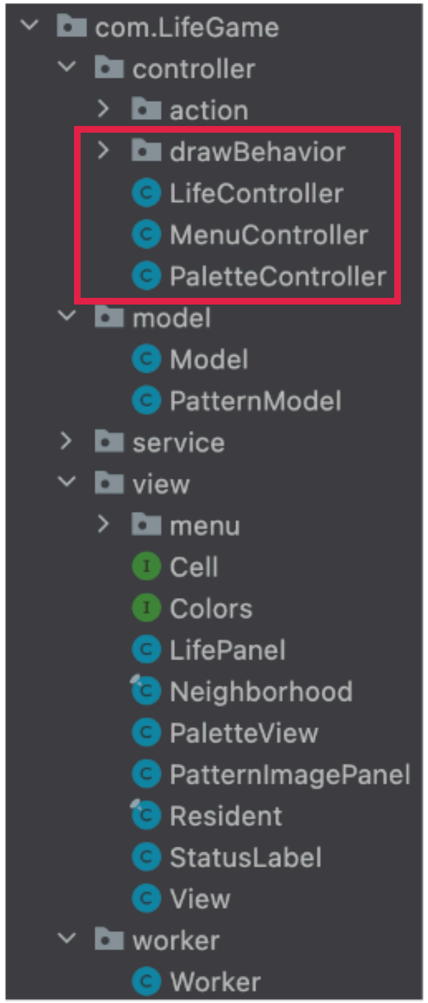
LifeGame – LifeController.java [LifeGame.main]

```java
2 usages  AllyHyeseongKim
public void setDrawBehavior(DrawBehavior drawBehavior) { this.drawBehavior = drawBehavior; }

3 usages  AllyHyeseongKim
public DrawBehavior getDrawBehavior() { return this.drawBehavior; }

2 usages  AllyHyeseongKim
public void addObserver(Observer o) { this.model.addObserver(o); }

2 usages  AllyHyeseongKim
public void mouseAction(int x, int y) {
    this.drawBehavior.draw(x, y);
}
```



Project Design

LifeGame – Model.java [LifeGame.main]

```java
@Component
public class Model extends Observable {

    6 usages
    private int size;
    17 usages
    private int[][] map;
    3 usages
    private int generation;

    12 usages  AllyHyeseongKim
    public void mapChanged() {
        setChanged();
        notifyObservers();
    }

    8 usages  AllyHyeseongKim +1
    public void clearMap() {  //map 초기화
        this.map = new int[this.size][this.size];
        this.generation = 0;
        this.mapChanged();
    }
```

LifeGame – Model.java [LifeGame.main]

```java
    13 usages  박서현 +1
    public void toggle(int x, int y) {
        // 좌표 설정 방식 - 그냥 배열이랑
        if (this.map[x][y] == 0) {
            this.map[x][y] = 1;
        } else {
            this.map[x][y] = 0;
        }

        this.mapChanged();
    }

    9 usages  AllyHyeseongKim +1
    public void setMap(int[][] map) {
        this.map = map;

        this.mapChanged();
    }

    AllyHyeseongKim +1
    public void draw(int[][] bluePri
        for (int i = 0; i < Math.min
            System.arraycopy(bluePri
        }

        this.mapChanged();
    }
```

LifeGame – PatternModel.java [LifeGame.main]

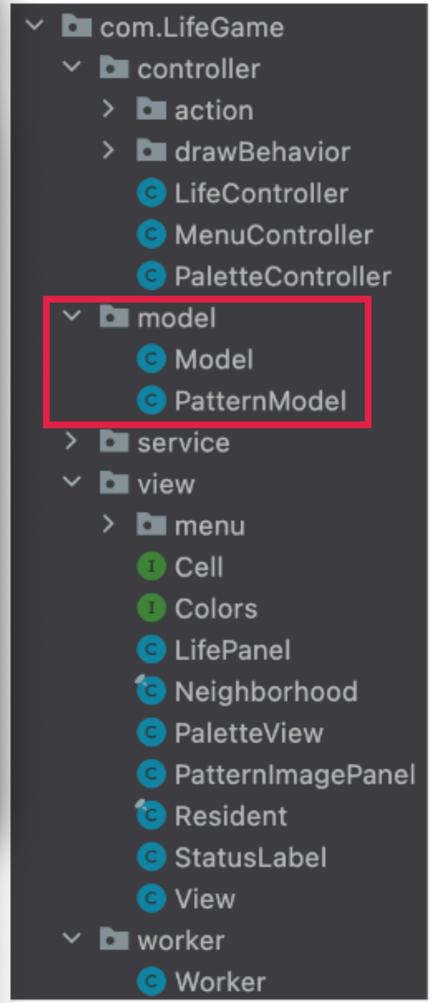
```java
@Component
@Setter @Getter
public class PatternModel extends Observable {

    3 usages
    private int id = 0;
    1 usage
    private int[][] bluePrint;

    4 usages  AllyHyeseongKim
    public void patternChanged() {
        setChanged();
        notifyObservers();
    }

    4 usages  AllyHyeseongKim
    public void changePattern(int id, int[][] bluePrint) {
        if (this.id == id) {
            this.id = 0;
        } else {
            this.id = id;
            this.bluePrint = bluePrint;
        }

        this.patternChanged();
    }
```



Project Design

LifeGame – AgonizingAction.java [LifeGame.main]

```java
AllyHyeseongKim
@Autowired
public AgonizingAction(Worker worke

AllyHyeseongKim
@Override
public void action() {
    this.worker.setSpeed(500);
    this.worker.startThread();
}
}
```

LifeGame – Worker.java [LifeGame.main]

```java
public void setSpeed(int speed) { this.speed = spe

AllyHyeseongKim
public void startThread() {
    if (this.thread != null && this.thread.isAlive()) {
        this.thread.interrupt();
    }

    AllyHyeseongKim
    this.thread = new Thread(new Runnable() {
        AllyHyeseongKim
        @Override
        public void run() {
            try {
                while (true) {
                    model.nextState();
                    sleep(speed);
                }
            } catch (InterruptedException e) {
            }
        }
    });
    this.thread.start();
```

- com.LifeGame
  - controller
    - action
    - drawBehavior
    - LifeController
    - MenuController
    - PaletteController
  - model
    - Model
    - PatternModel
  - service
  - view
    - menu
    - Cell
    - Colors
    - LifePanel
    - Neighborhood
    - PaletteView
    - PatternImagePanel
    - Resident
    - StatusLabel
    - View
  - worker
    - Worker

Implementation

Test Results — 5 sec 554 ms

- AgonizingActionTest — 33 ms
- ClearActionTest — 2 ms
- DefaultDrawBehaviorTest
- FastActionTest
- HaltActionTest
- LifeControllerTest
  - [mouseAction] draw 메서드가 잘 호출되는지 테스트
  - [update/default] 업데이트 상황에 따라 setDrawBehavior 메서드가 잘 호출되는지 테스트
  - [update/pattern] 업데이트 상황에 따라 setDrawBehavior 메서드가 잘 호출되는지 테스트
- LifeGameApplicationTests
- LifePanelTest
- LoadActionTest
- MediumActionTest
- MenuBarTest
- MenuControllerTest
- MenuTest
- ModelTest
- PaletteControllerTest
- PatternDrawBehaviorTest
- PatternImagePanelTest
- PatternModelTest
- ServiceTest
- SlowActionTest
- StatusLabelTest
- StoreActionTest
- TickActionTest
- WorkerTest

LifeGame – LifeControllerTest.java [LifeGame.test]

LifeGameApplication | Git:

LifeControllerTest.java

AllyHyeseongKim

```
19  @ExtendWith(MockitoExtension.class)
20  class LifeControllerTest {
21
22      @Mock
23      private Model model;
        4 usages
24      @Mock
25      private PatternModel patternModel;
26
        1 usage
27      @Mock
28      private DefaultDrawBehavior defaultDrawBehavior;
        1 usage
29      @Mock
30      private PatternDrawBehavior patternDrawBehavior;
31
        6 usages
32      @InjectMocks
33      private LifeController lifeController;
34
```

AllyHyeseongKim

Project | Commit | Pull Requests | Bookmarks | Structure | Gradle | Database | Big Data Tools | Notifications

Git | Run | TODO | Problems | Spring | Terminal | Endpoints | Services

76:18 (61 chars, 1 line break) LF UTF-8 4 spaces master

Implementation

LifeGame [~/workspaces/projects/LifeGame] – gradle.yml [LifeGame]

LifeGame 〉 .github 〉 gradle.yml

```yaml
name: Java CI with Gradle

on:
  push:
    branches: [ "**" ]
  pull_request:
    branches: [ "**" ]

jobs:
  unit_test:

    runs-on: ubuntu-latest

    steps:
      - uses: actions/checkout@v3
      - name: Set up JDK 11
        uses: actions/setup-java@v3
        with:
          java-version: '11'
          distribution: 'corretto'

      - name: Build with Gradle
        uses: gradle/gradle-build-action@67421db6bd0bf253fb4bd25b31ebb98943c375e1
        with:
          arguments: build

      - name: Grant execute permission for gradlew
        run: chmod +x gradlew

      - name: Test with Gradle
        run: ./gradlew --info test
```

LifeGameApplication: Failed to retrieve appli... (yesterday 7:16 PM) 1:1 LF UTF-8 2 spaces No JSON schema master

github.com

Search or jump to... / Pulls Issues Codespaces Marketplace Explore

CAU-DesignPattern / LifeGame Public

Edit Pins | Watch 0 | Fork 3 | Starred 2

Code | Issues | Pull requests | Actions | Projects 1 | Wiki | Security

Actions New workflow

All workflows

Java CI with Gradle

Management

Caches

All workflows

Showing runs from all workflows

Filter workflow runs

101 workflow runs

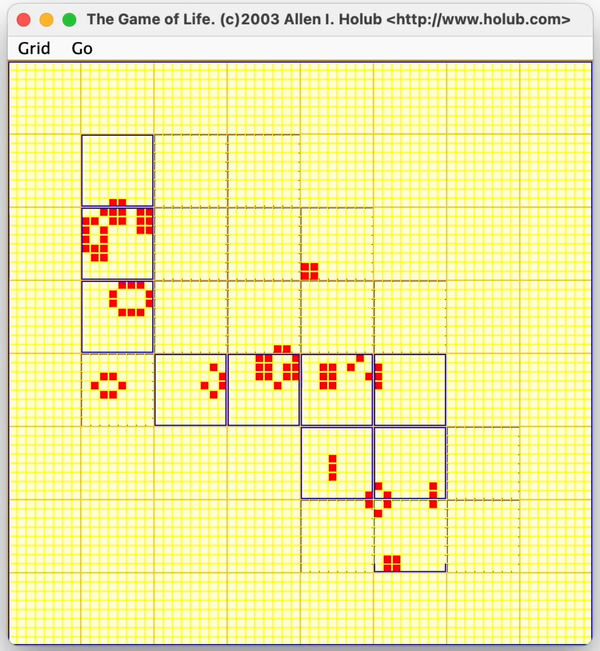
Event | Status | Branch | Actor

- **Merge pull request #47 from C...** — Java CI with Gradle #101: Commit 9aedad5 pushed by AllyHyeseongKim — master — 3 days ago — 53s
- **Merge 'develop' into master** — Java CI with Gradle #100: Pull request #47 opened by AllyHyeseongKim — develop — 3 days ago — 46s
- **Merge pull request #46 from C...** — Java CI with Gradle #99: Commit 8dc7495 pushed by AllyHyeseongKim — develop — 3 days ago — 49s
- **Merge 'feature/view' into develop** — Java CI with Gradle #98: Pull request #46 opened by AllyHyeseongKim — feature/view — 3 days ago — 51s
- **[feature/view] Delete Observable** — Java CI with Gradle #97: Commit 6a9bbe3 — feature/view — 3 days ago

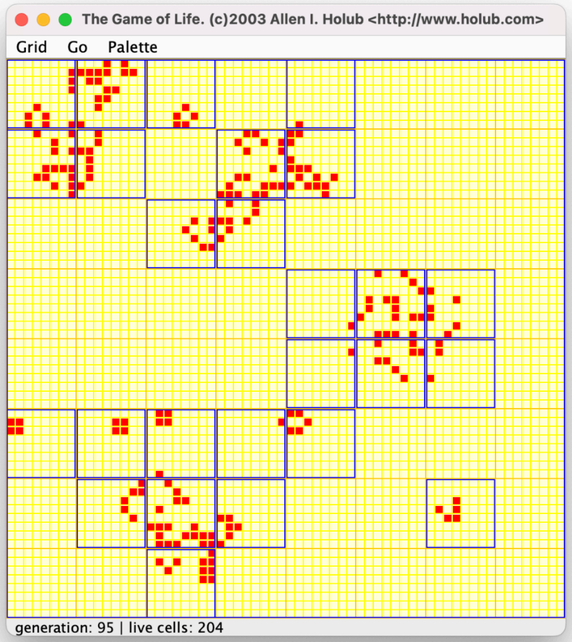
Results

■ 설계 개선 장면

MVC 패턴 적용 설계 결과 neighborhood와 파란색 테두리 올바르게 그려짐

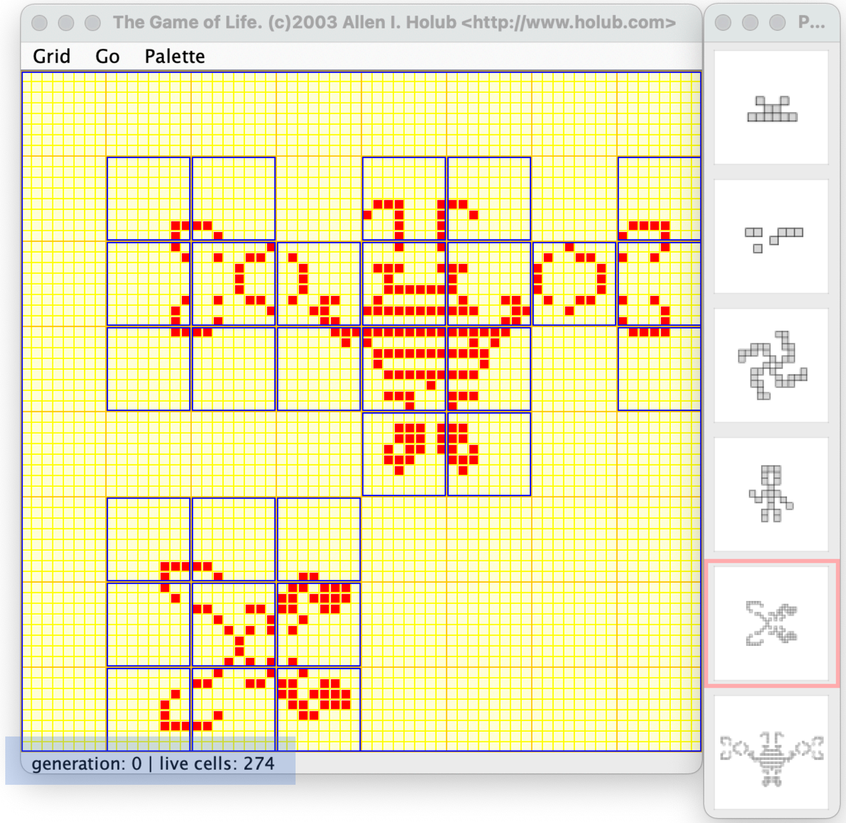


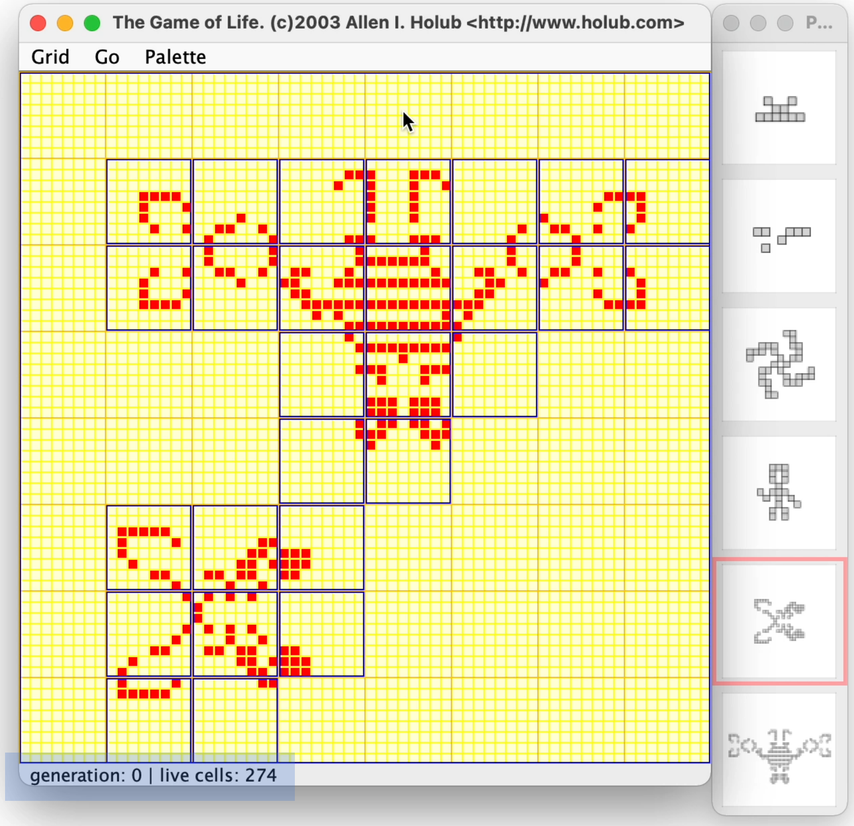
→



Results

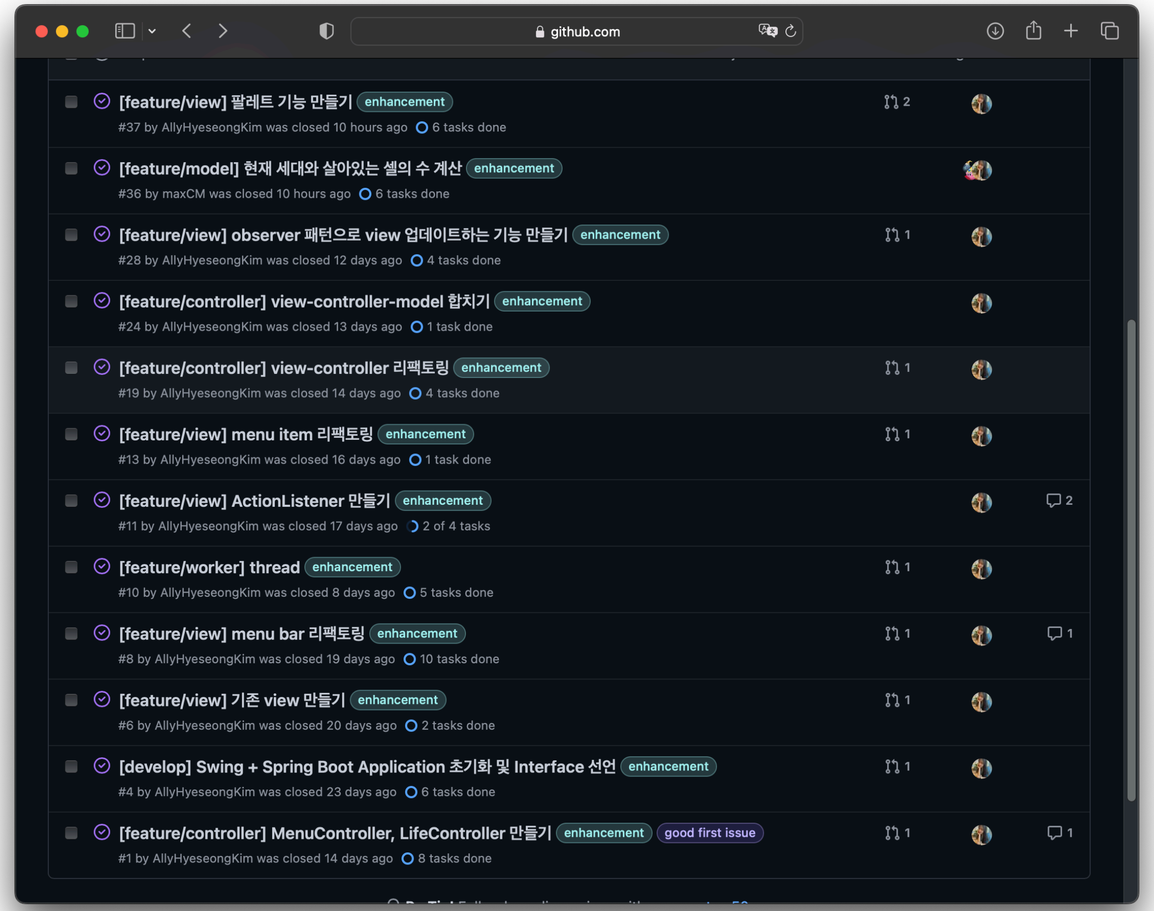
■ 확장 기능 동작 장면

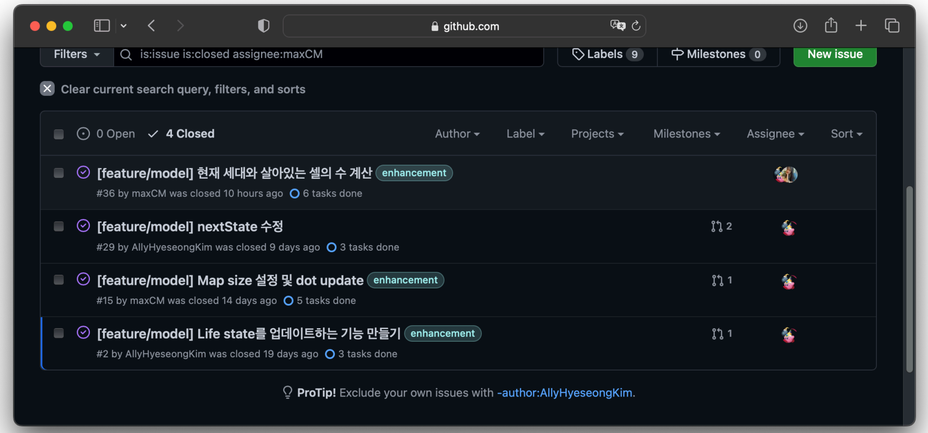


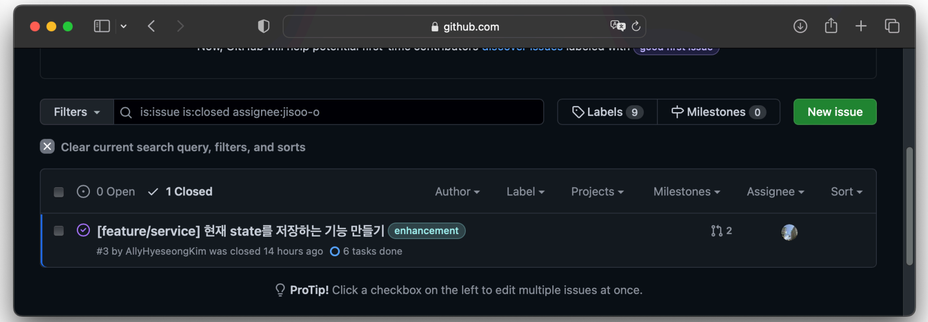


Results

■ 협업 방법







Thank You 🤗