МИНИСТЕРСТВО НАУКИ И ВЫШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРТСВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ АВТОМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

**Кафедра вычислительной техники**



**Лабораторная работа №4**

**по предмету «Инжиниринг программного обеспечения»**

Графический интерфейс с динамической разметкой

Вариант 1

Группа: АВТ-143

Студенты: Чалов В.П., Комаров П.В.

Преподаватель: Романов Е. Л.

Новосибирск 2024

Содержание:

[Результат: 3](#_Toc176907490)

[Задача: 3](#_Toc176907491)

[Описание: 3](#_Toc176907492)

[UML диаграмма: 3](#_Toc176907493)

[Моделирование 4](#_Toc176907494)

[Интерфейс: 8](#_Toc176907495)

[Устройство приложения изнутри: 9](#_Toc176907496)

[Известные проблемы устройства приложения: 14](#_Toc176907497)

[Вывод: 14](#_Toc176907498)

Результат:[Ссылка на приложение](https://github.com/peachyhead/pi_4/tree/master)

# Задача:

Физический объект – кастрюля на плите: текущее положение конфорки, температура и объем содержимого, состояние нагревания и кипения, теплоотдача при открытой и закрытой крышке.

# Описание:

Класс-контроллер реализует в отдельном потоке в реальном времени модель поведения физического объекта. Классы GUI (формы) получают ссылку на объект-контроллер с интерфейсом команд и подписываются на получение от него событий через соответствующий интерфейс (шаблон наблюдатель). При появлении в контроллере события последний вызывает соответствующий метод во всех подписанных объектах. При закрытии объекта класса GUI он отписывается от контроллера. События и команды в интерфейсах должны обеспечивать объектам внешнего представления полный набор возможностей управления и отображения состояния физического объекта.

# UML диаграмма:

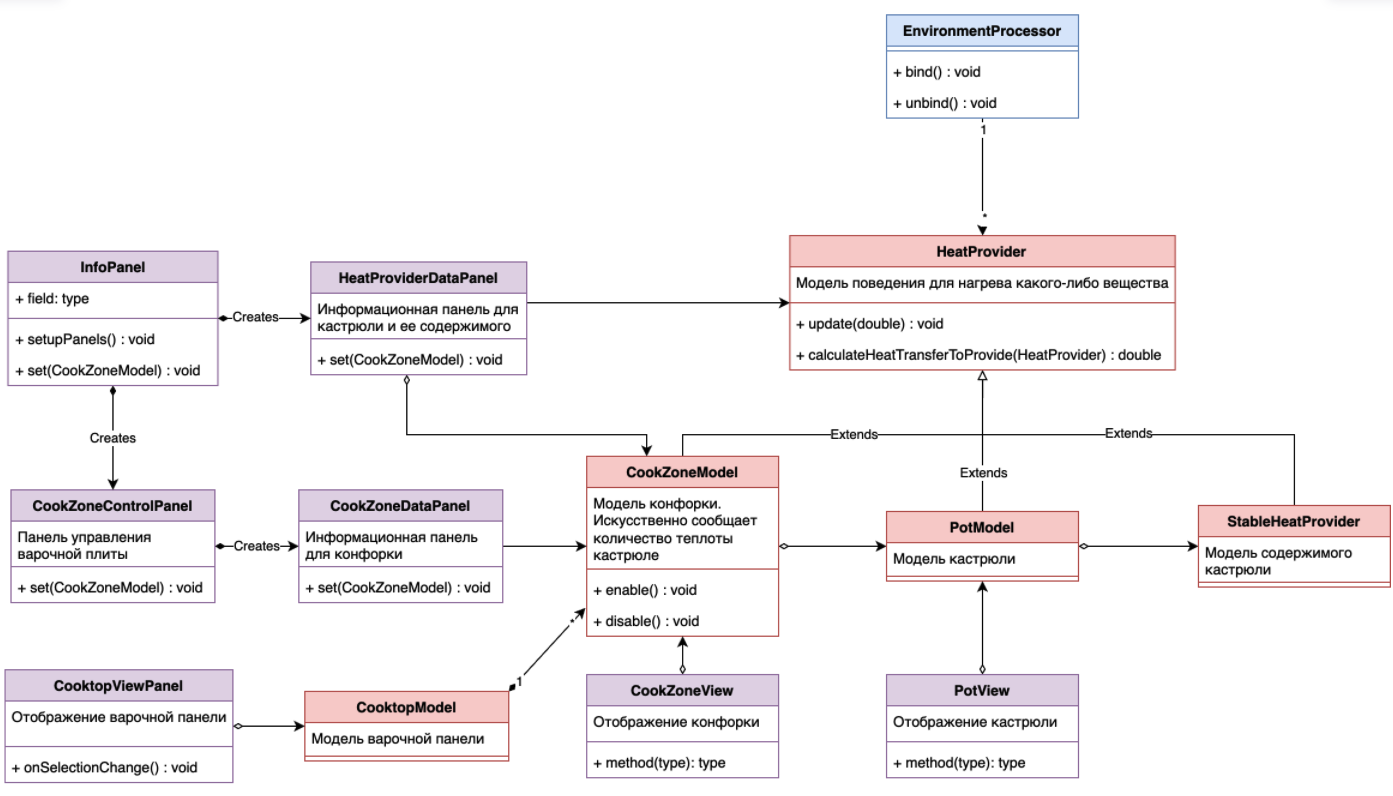


Рисунок 1. Поверхностная uml диаграмма приложения.

На диаграмме вынесен жизненный цикл в отдельный модуль (каковым он не является), чтобы улучшить читаемость.

*Определение сущностей диаграммы:*

**красный** - модель данных

**фиолетовый** – отображение

**голубой** - сервис

Не станем упоминать на диаграмме вспомогательные инструменты (SignalBus, SignalListener). Можно назвать это сахаром, который делает вещи удобнее.

# Моделирование

Для создания необходимой симуляции, нужно учитывать:

- передачу тепла от конфорки к кастрюле

- передачу тепла от кастрюли к ее содержимому

- теплообмен с окружающей средой

- состояния нагрева жидкости

Мы будем использовать следующие принципы термодинамики:

**1. Закон теплопередачи (уравнение теплопередачи)**

Теплопередача между двумя объектами или между объектом и окружающей средой рассчитывается по формуле:

Где:

• Q — количество тепла, переданное между объектом и окружающей средой (Дж).

• k — коэффициент теплопередачи (Вт/м²°C).

• S — площадь поверхности теплопередачи (м²).

•  — разность температур между объектом и окружающей средой

**2. Изменение температуры в результате поглощенного тепла**

Температура объекта меняется в зависимости от того, сколько тепла он поглощает. Это рассчитывается по формуле:

Где:

• — изменение температуры объекта (°C).

• Q — количество тепла, поглощенное объектом (Дж).

• C — удельная теплоемкость объекта (Дж/кг°C).

• m — масса объекта (кг).

Температура объекта обновляется по формуле:

**3. Масса объекта**

Масса объекта (в данном случае — жидкости) рассчитывается как:

Где:

• m — масса объекта (кг).

• V — объем объекта (м³).

• — плотность объекта (кг/м³).

**4. Испарение жидкости**

При нагреве жидкости до точки кипения она начинает испаряться. Количество тепла, необходимое для испарения некоторого объема жидкости, рассчитывается по формуле:

Где:

• — количество тепла, необходимое для испарения объема жидкости (Дж).

• L — удельная теплота испарения (Дж/кг).

• m — масса испаряющейся жидкости (кг).

Чтобы определить объем испарившейся жидкости, используется формула:

Где:

• — объем испарившейся жидкости (м³).

• — количество тепла, поглощенное жидкостью (Дж).

• L — удельная теплота испарения (Дж/кг).

**5. Теплоемкость жидкости**

Теплоемкость жидкости рассчитывается с использованием удельной теплоемкости (С) и массы жидкости:

Где:

• — общая теплоемкость жидкости (Дж/°C).

• C — удельная теплоемкость жидкости (Дж/кг°C).

• m — масса жидкости (кг).

Это значение используется для расчета того, насколько сильно изменится температура жидкости при поглощении или отдаче тепла.

**6. Общая формула расчета температуры жидкости**

При нагреве жидкости из окружающей среды температура жидкости обновляется по следующей формуле:

Где:

• — новая температура жидкости после теплового обмена (°C).

• — текущая температура жидкости (°C).

• — температура окружающей среды (°C).

• k — коэффициент теплопередачи (Вт/м²°C).

• S — площадь теплопередачи (м²).

• — общая теплоемкость жидкости (Дж/°C).

**7. Изменение температуры окружающей среды**

Тепловой обмен между жидкостью и окружающей средой также приводит к изменению температуры окружающей среды. Температура окружающей среды изменяется на величину, зависящую от общего теплового баланса между средой и объектом:

Где:

• — изменение температуры окружающей среды (°C).

• — общее количество тепла, переданное жидкостью или другими объектами в окружающую среду (Дж).

• — теплоемкость окружающей среды (Дж/°C).

**8. Уравнение теплового баланса**

Для объектов в системе происходит непрерывный тепловой обмен, который подчиняется закону сохранения энергии:

# Интерфейс:

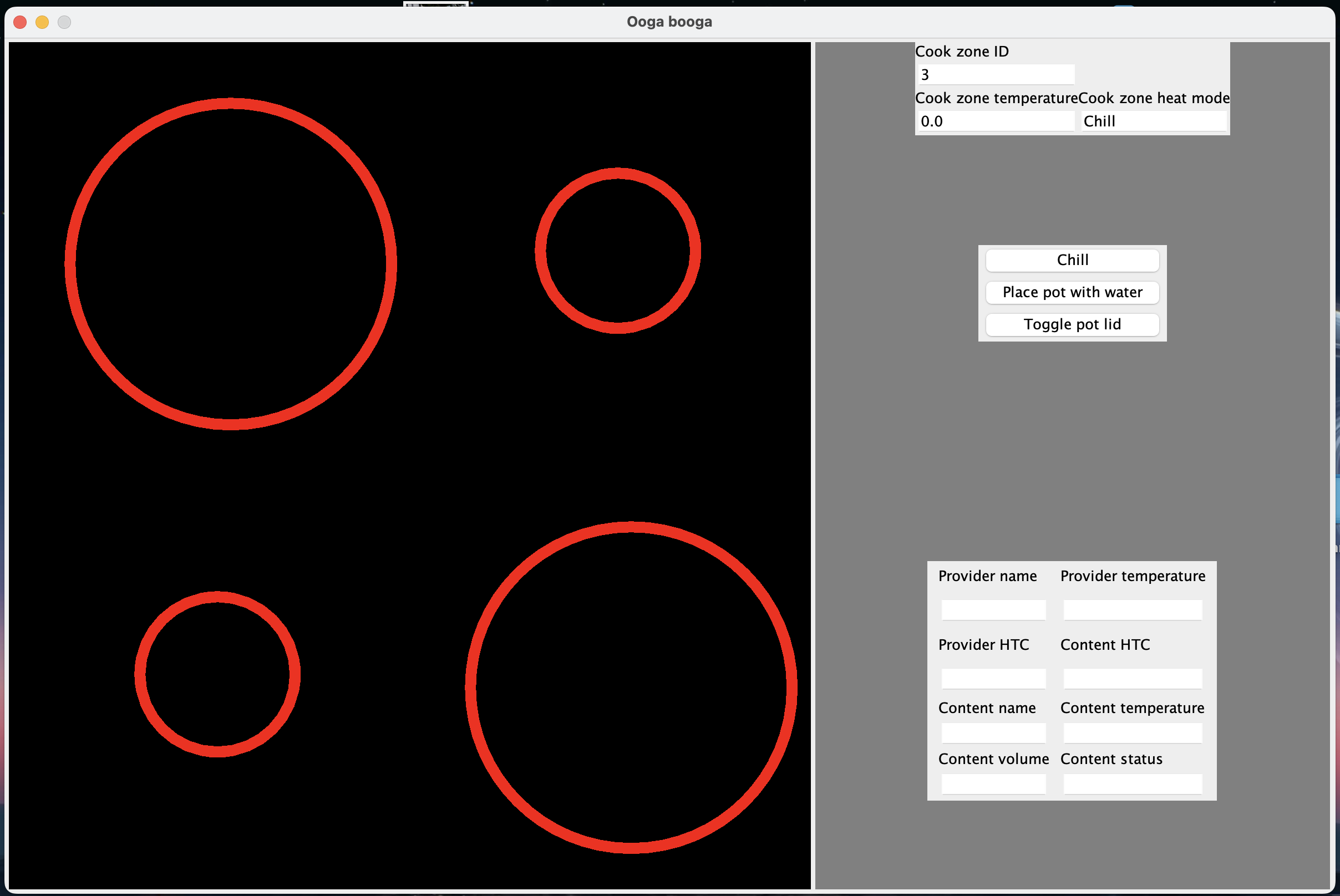


Рисунок 2. Интерфейс приложения

При клике на конфорку, можно выбрать режим нагрева, поставить/убрать кастрюлю, положить/убрать крышку с кастрюли

Информационная панель справа разделена на две части – сверху отображена информация о конфорке (температура, id, режим нагрева). Снизу отображена информация о том, что находится на конфорке.

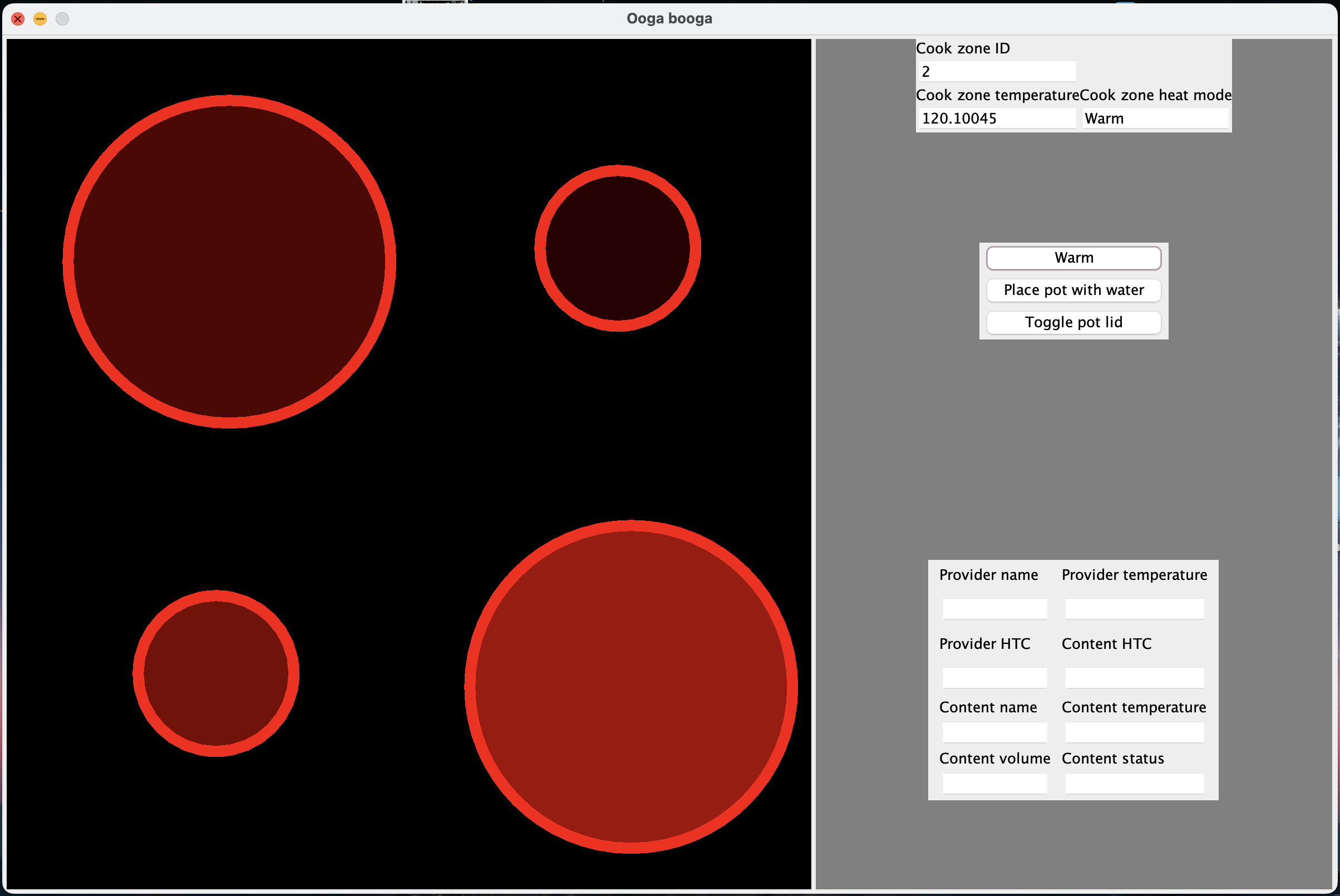


Рисунок 3. Функционирование конфорок плиты

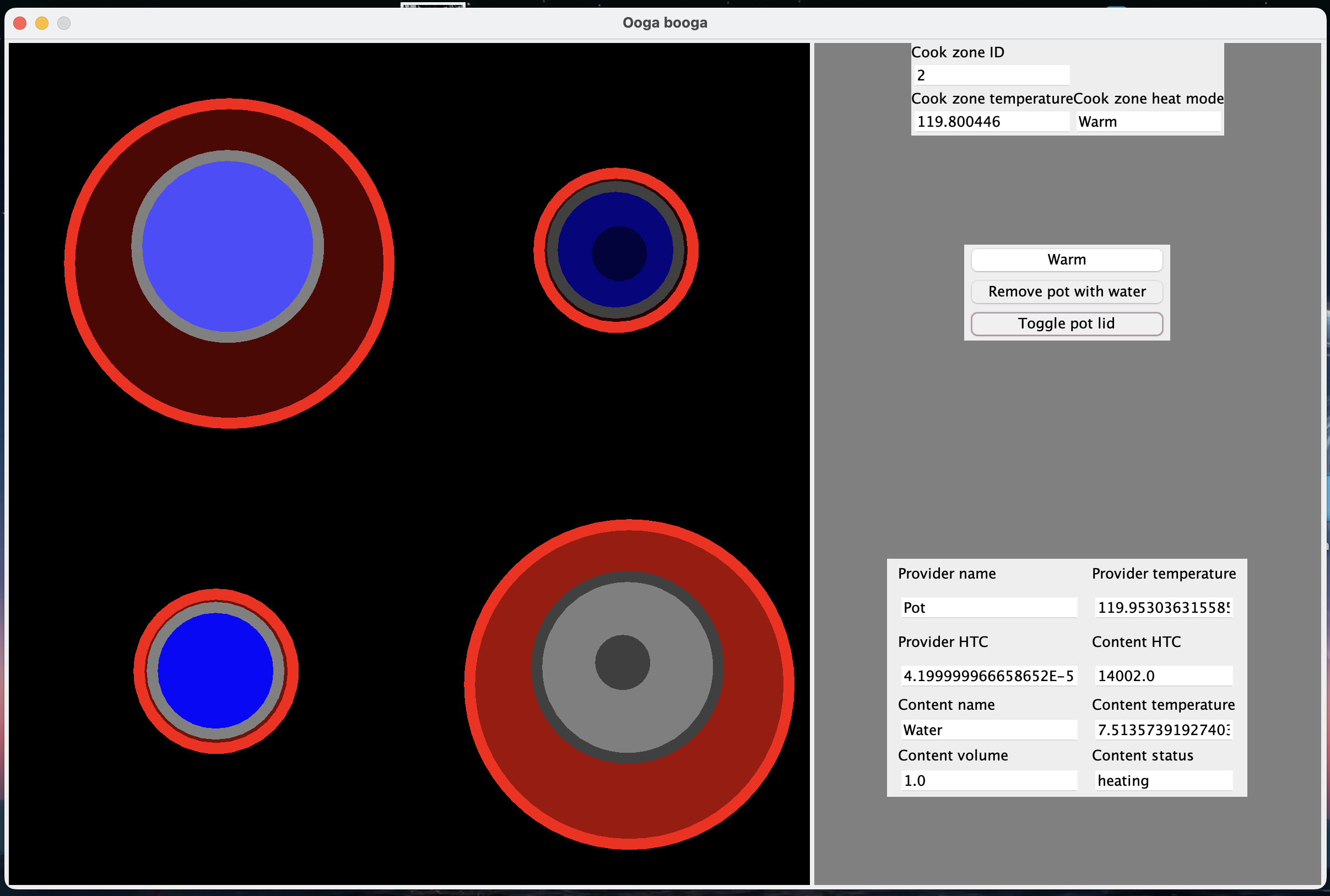


Рисунок 4. Отображение кастрюли на плите (с крышкой и без)

# Устройство приложения изнутри:

**1. Модуль конфорки (CookZoneModel)**

* **Управление режимом нагрева:** Конфорка имеет несколько режимов нагрева. Режимы влияют на количество теплоты, которое сообщаются объекту на конфорке.
* **Нагрев:** Когда конфорка активирована, она создает новый поток (heatTransferStream), который периодически обновляет текущую тепловую передачу. Эта тепловая передача зависит от режима нагрева и текущего состояние тепловой передачи (currentHeatTransfer).
* **Тепловая передача:** Конфорка использует базовую теплопередачу (baseHeatTransfer) и текущий режим для вычисления целевого уровня тепла (getTargetHeatTransfer). Затем она постепенно увеличивает текущую теплопередачу до достижения этого уровня.



Рисунок 5. Поток для плавного сообщения тепла объекту на конфорке

**2. Модуль источника тепла (HeatProvider)**

Вся логика нагрева и перехода между состояниями находится в HeatProvider

* **Основные параметры:** Это абстрактный класс, который содержит базовые параметры для источника тепла, такие как плотность, объем, удельная теплоемкость и т.д. А также логику нагрева и перехода между тепловыми состояниями.
* **Обновление температуры:** HeatProvider имеет метод handleHeatAbsorb, который обновляет температуру объекта в зависимости от полученного тепла и его свойств. Если температура достигает точки кипения, объект начинает испаряться.
* **Тепловой обмен:** HeatProvider может передавать тепло другому источнику тепла через метод update(). Метод calculateHeatTransferToProvider() учитывает разницу температур и площадь поверхности для расчета передаваемого тепла.

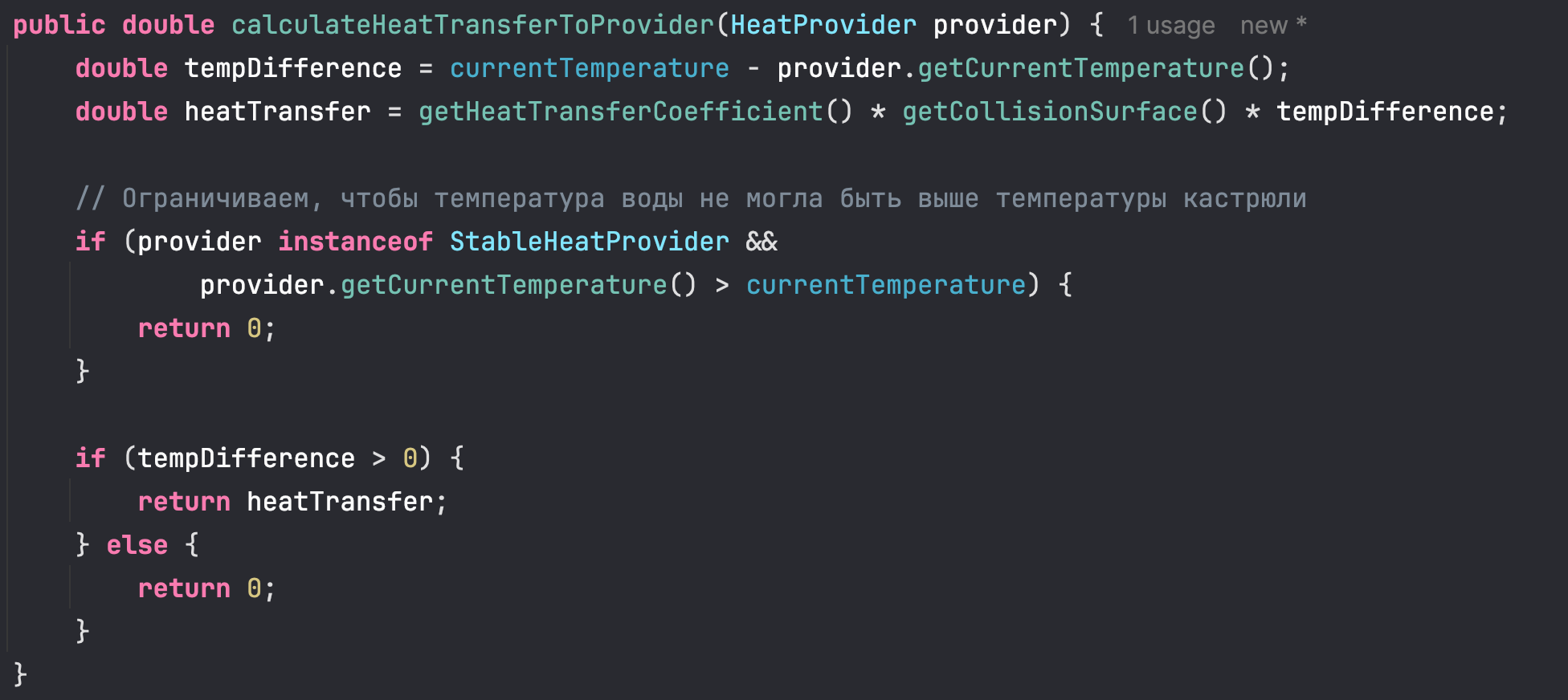


Рисунок 6. Расчет количества тепла для сообщения другому объекту

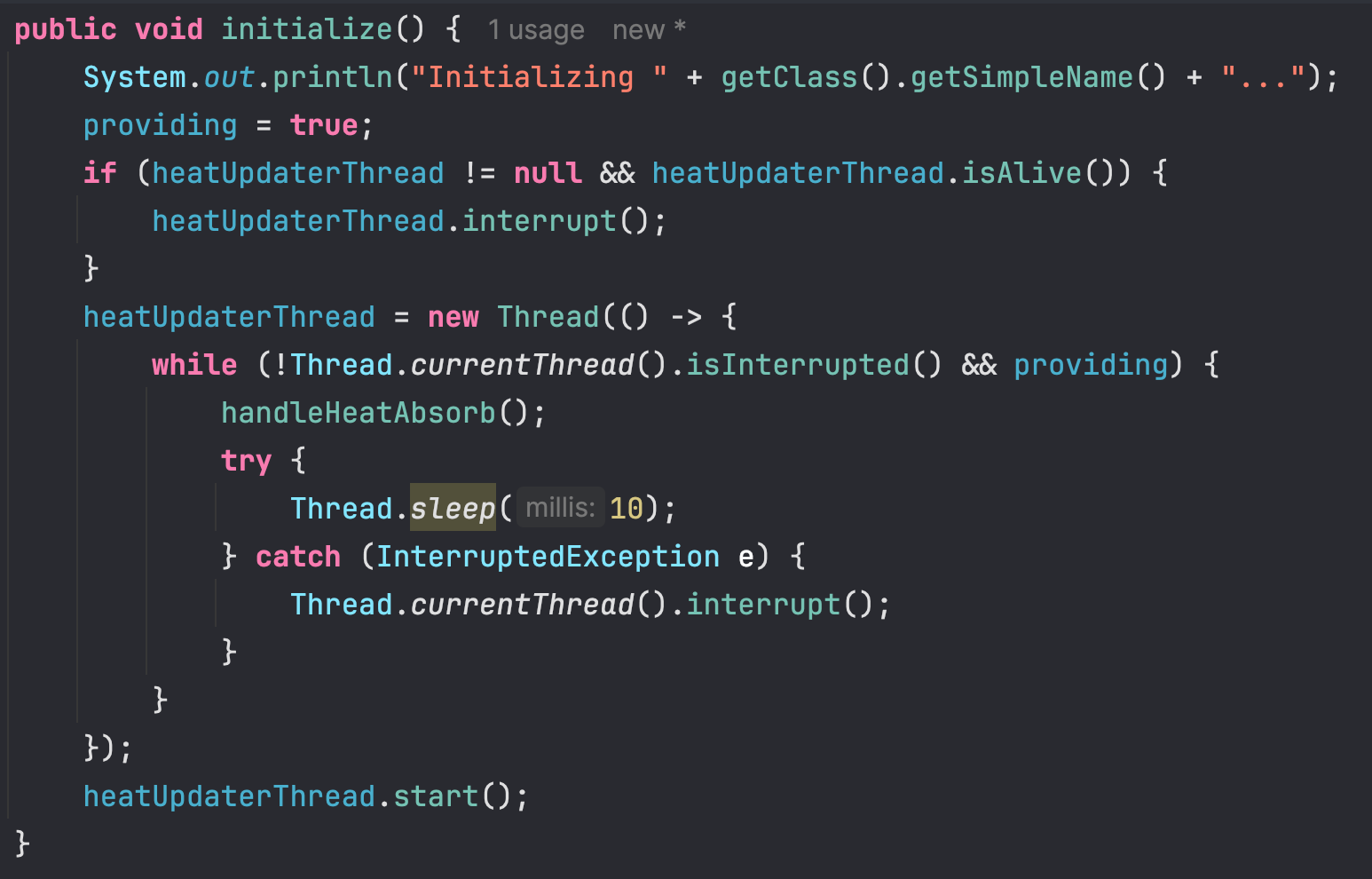


Рисунок 7. Поток для обновления температуры объекта HeatProvider

# 

Рисунок 8. Расчет температуры объекта от количества тепла, которое ему сообщили

за тик (100 мс)

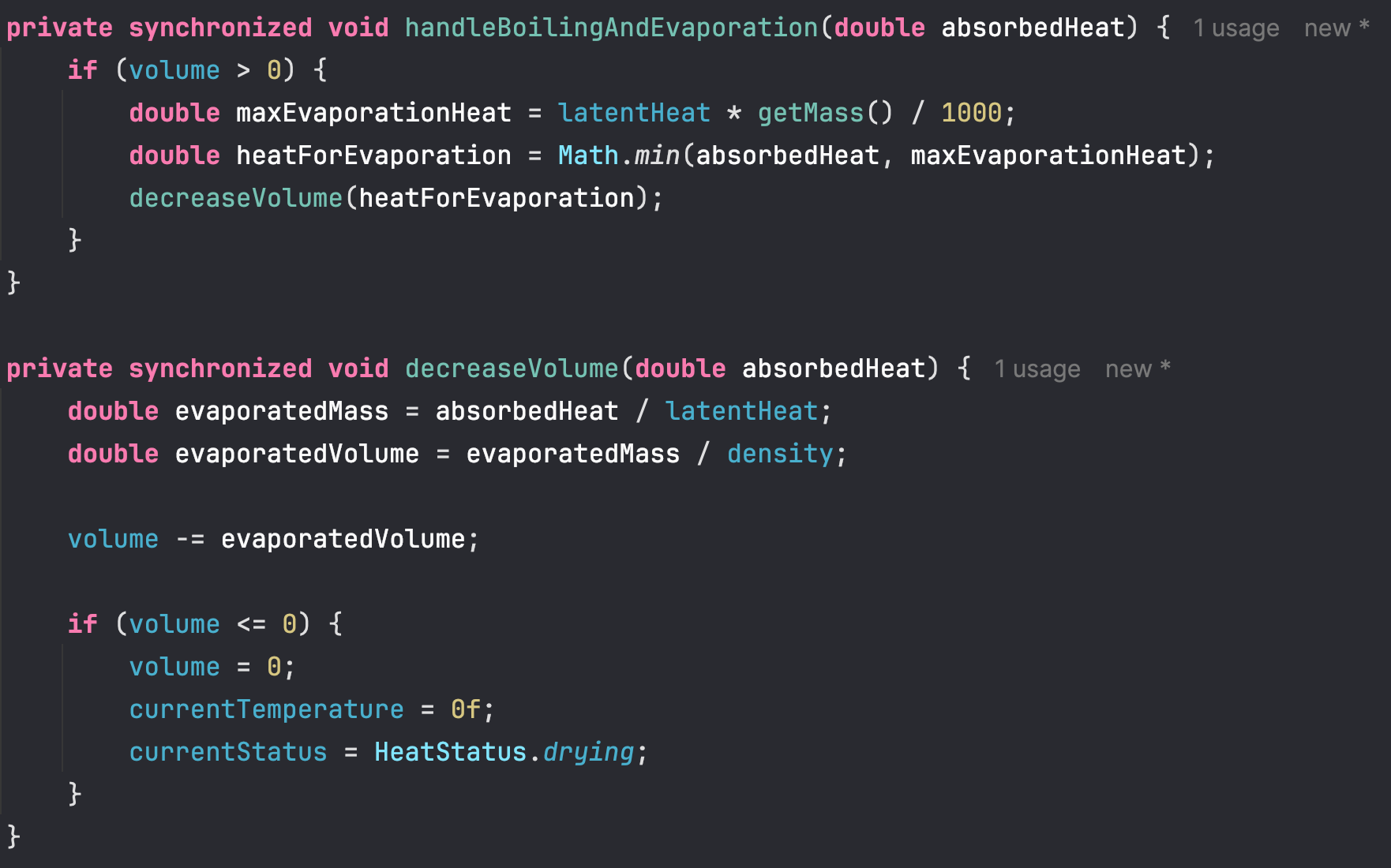
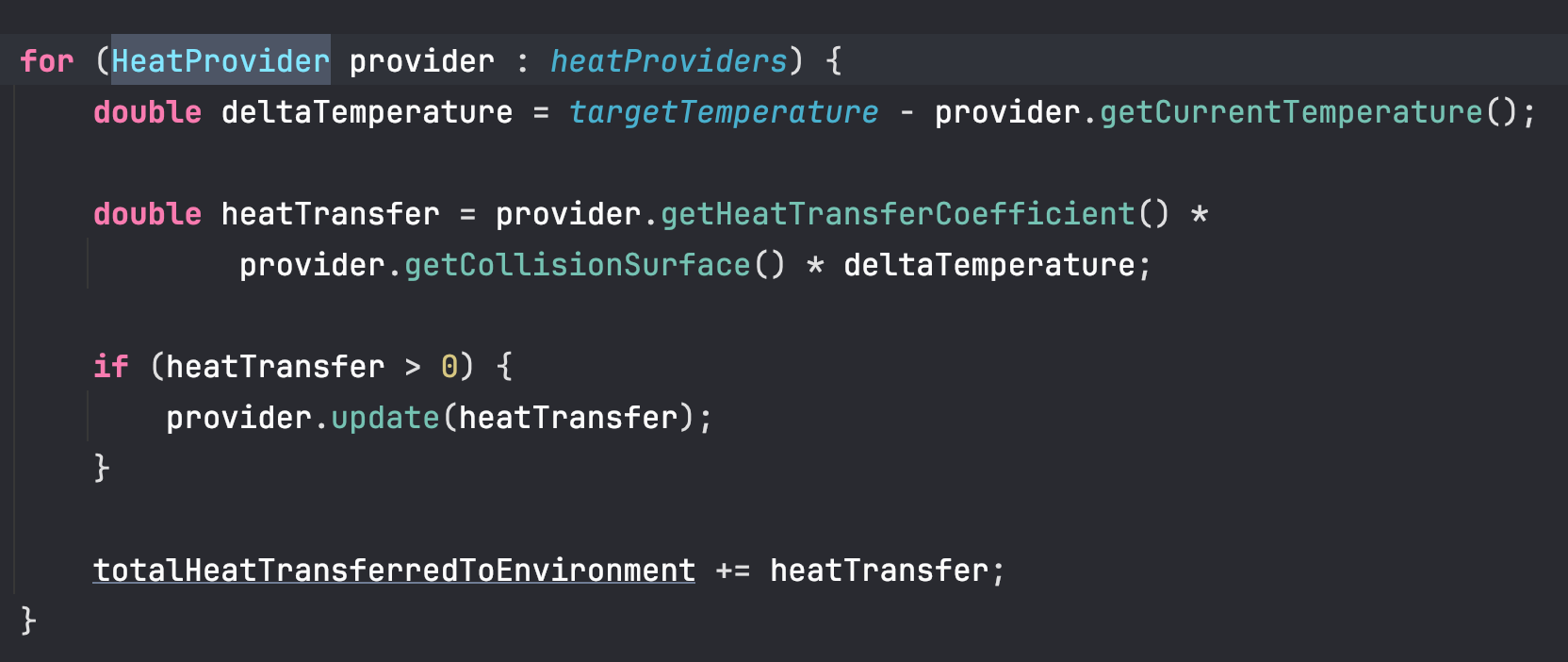


Рисунок 9. Логика испарения вещества

**3. Модуль обработки среды (EnvironmentProcessor)**

* **Тепловой обмен с окружающей средой:** EnvironmentProcessor управляет тепловым обменом между источниками тепла и окружающей средой. Он вычисляет, сколько тепла передается в окружающую среду и обновляет целевую температуру среды (targetTemperature).
* **Обмен теплом между источниками:** Он также управляет обменом теплом между разными источниками тепла, используя exchangeMap. В этом случае, если pot (кастрюля) и water (вода) связаны, EnvironmentProcessor вычисляет и обновляет тепло, передаваемое между ними.



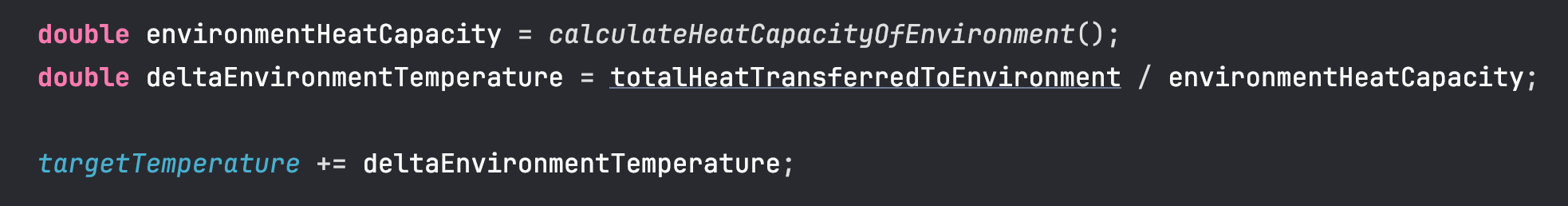


Рисунок 10. Логика теплообмена с окружающей средой

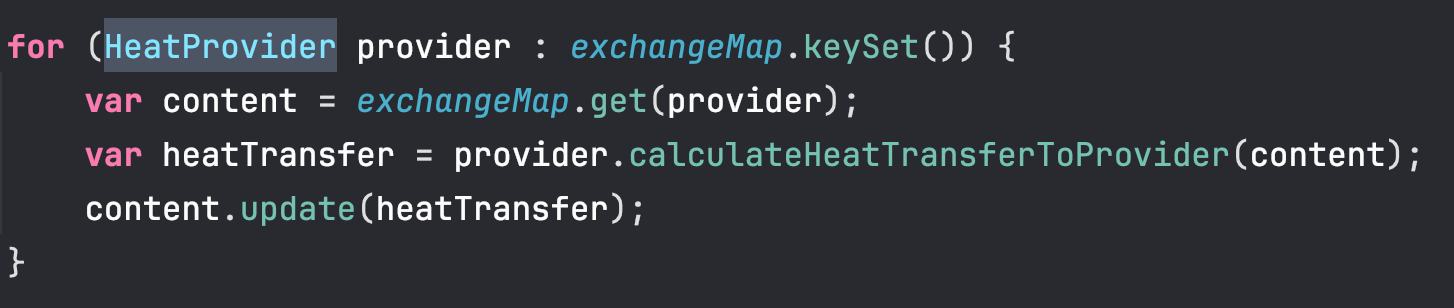


Рисунок 11. Логика теплообмена между объектами

Для настройки теплообмена, объект нужно “зарегистрировать” в EnvironmentProcessor.



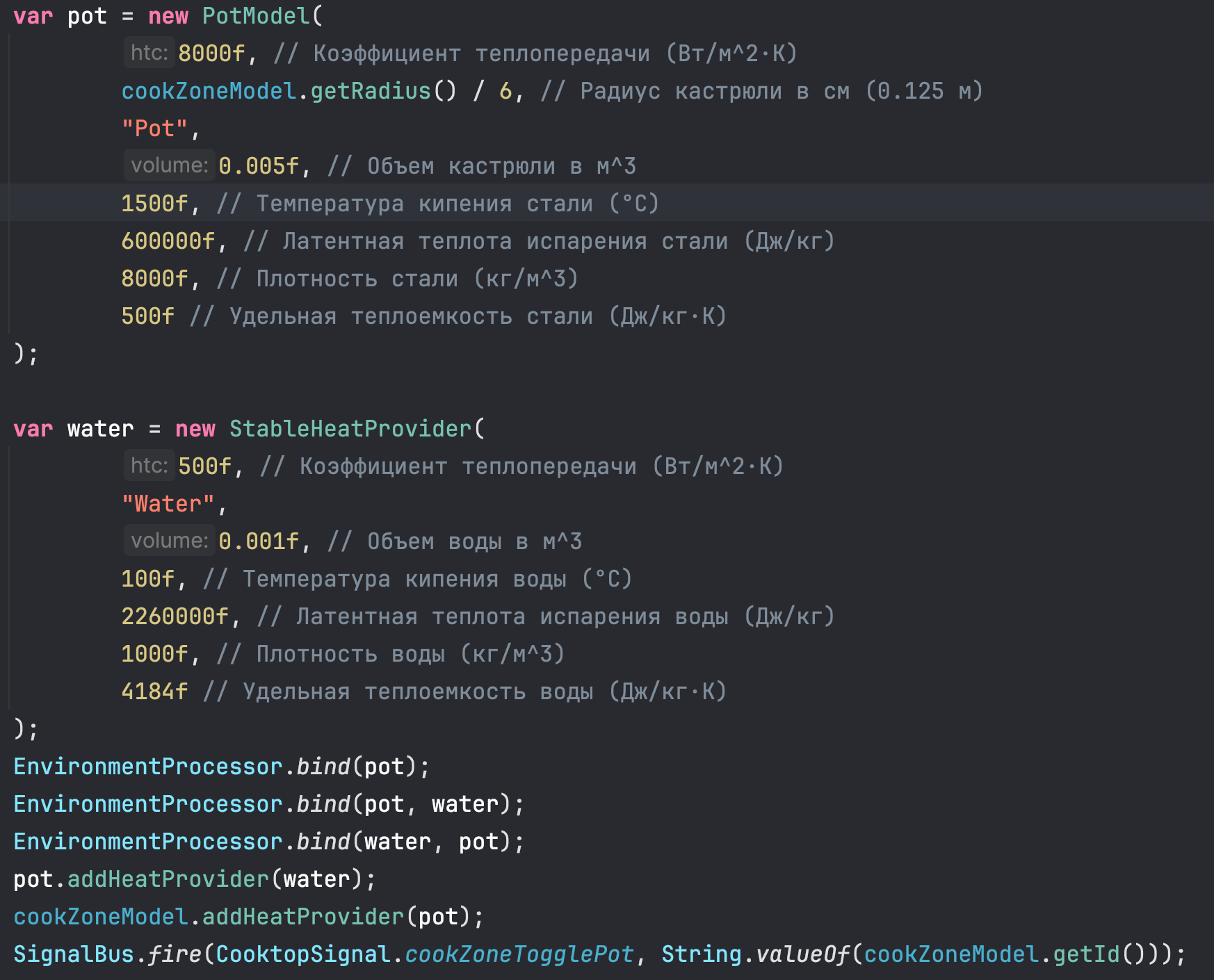


Рисунок 12. Регистрация объекта в EnvironmentProcessor

**4. Интерфейс для получения событий конфорки:**

Для избежания сильной связанности при получении событий, был введен классы SignalBus и SignalListener. Суть в том, чтобы посылать сигналы из любой точки кода, и подписываться на них в нужных местах. Вот, как выглядит запуск сигнала:

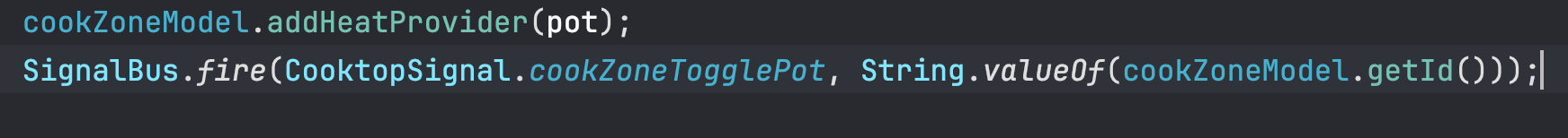


Рисунок 13. Запуск сигнала о добавлении/удалении кастрюли с конфорки (CookZoneControlPanel)



Рисунок 14. Запуск сигнала о смене режима нагрева конфорки (CookZoneControlPanel)

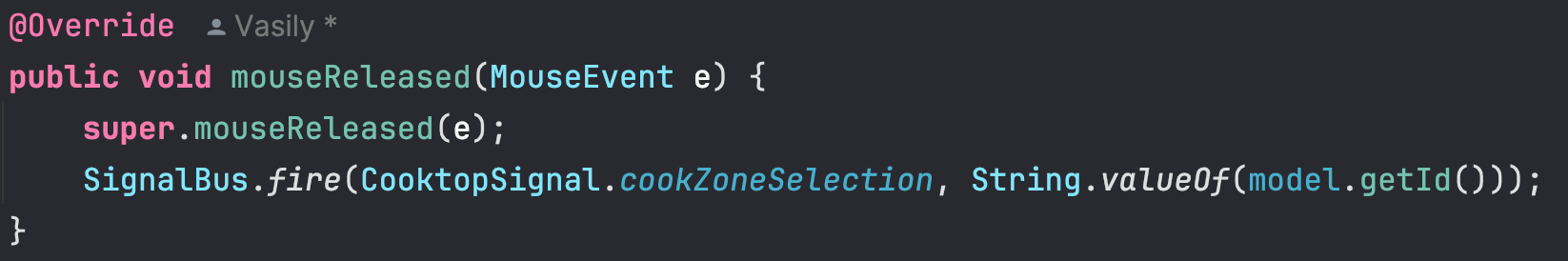


Рисунок 15. Запуск сигнала смены выбранной конфорки (CookZoneView)

Для того, чтобы подписаться на сигнал, нужно получить его литерал из статического класса, и определить PropertyChangeListener:

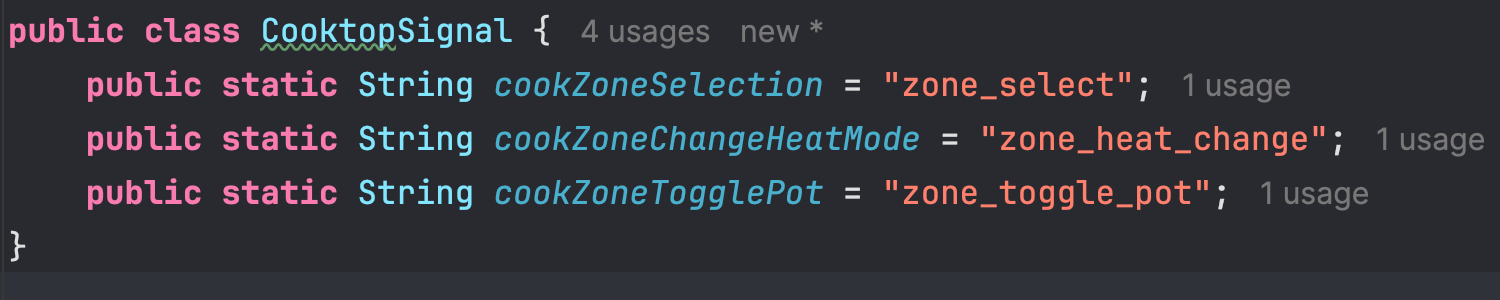


Рисунок 16. Декларирование сигналов конфорки

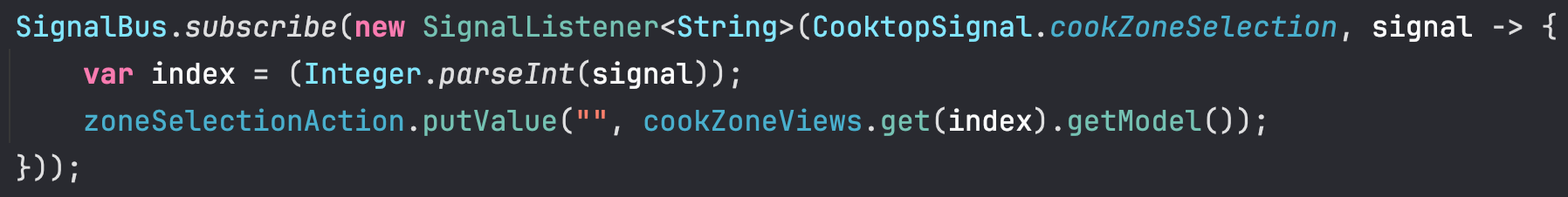


Рисунок 17. Подписка на сигнал смены выбранной конфорки (CooktopViewPanel)

После оповещения о действии, можно обратиться к CooktopModel, чтобы получить нужную конфорку по id из сигнала

# Известные проблемы устройства приложения:

1. Много логики в отображении. Можно было грамотнее завязать подписки, тем самым сняв лишнюю ответственность с отображений
2. Применение «магических цифр». Для упрощения тестирования и проверки работы, были использованы определенные константы в физических расчетах. Жертва точности симуляции в пользу времени разработки

# Вывод:

Следуя требованиям к заданию, была реализована симуляция варочной панели, которая включает в себя реагирование на действия пользователя и логику расчета передачи тепла между объектами