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:

[Задача: 2](#_Toc176729569)

[Описание: 3](#_Toc176729570)

[UML диаграмма: 3](#_Toc176729571)

[Интерфейс: 4](#_Toc176729572)

[Устройство приложения изнутри: 5](#_Toc176729573)

[Известные проблемы устройства приложения: 10](#_Toc176729574)

[Вывод: 10](#_Toc176729575)

# 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:

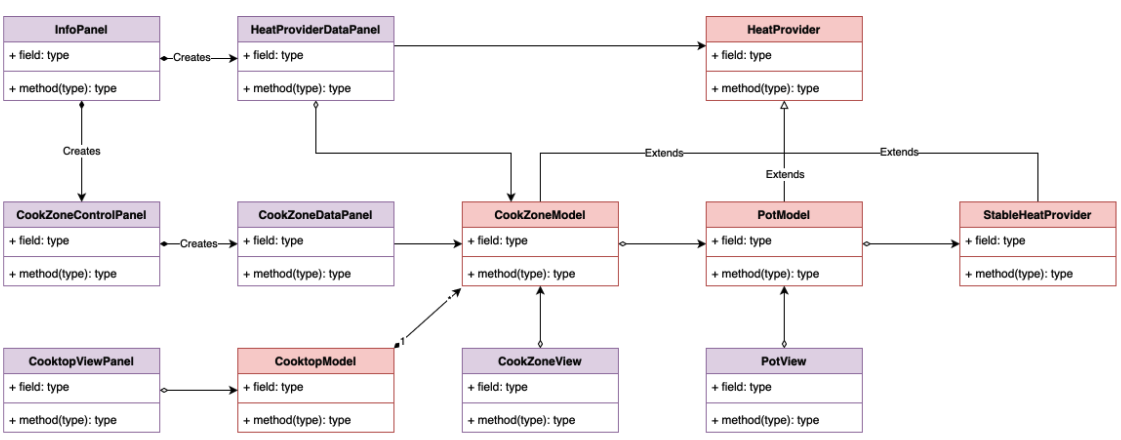


Рисунок 1. Поверхностная uml диаграмма приложения.

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

*Определение сущностей диаграммы:*

**красный** - модель данных

**фиолетовый** - отображение

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

# Интерфейс:

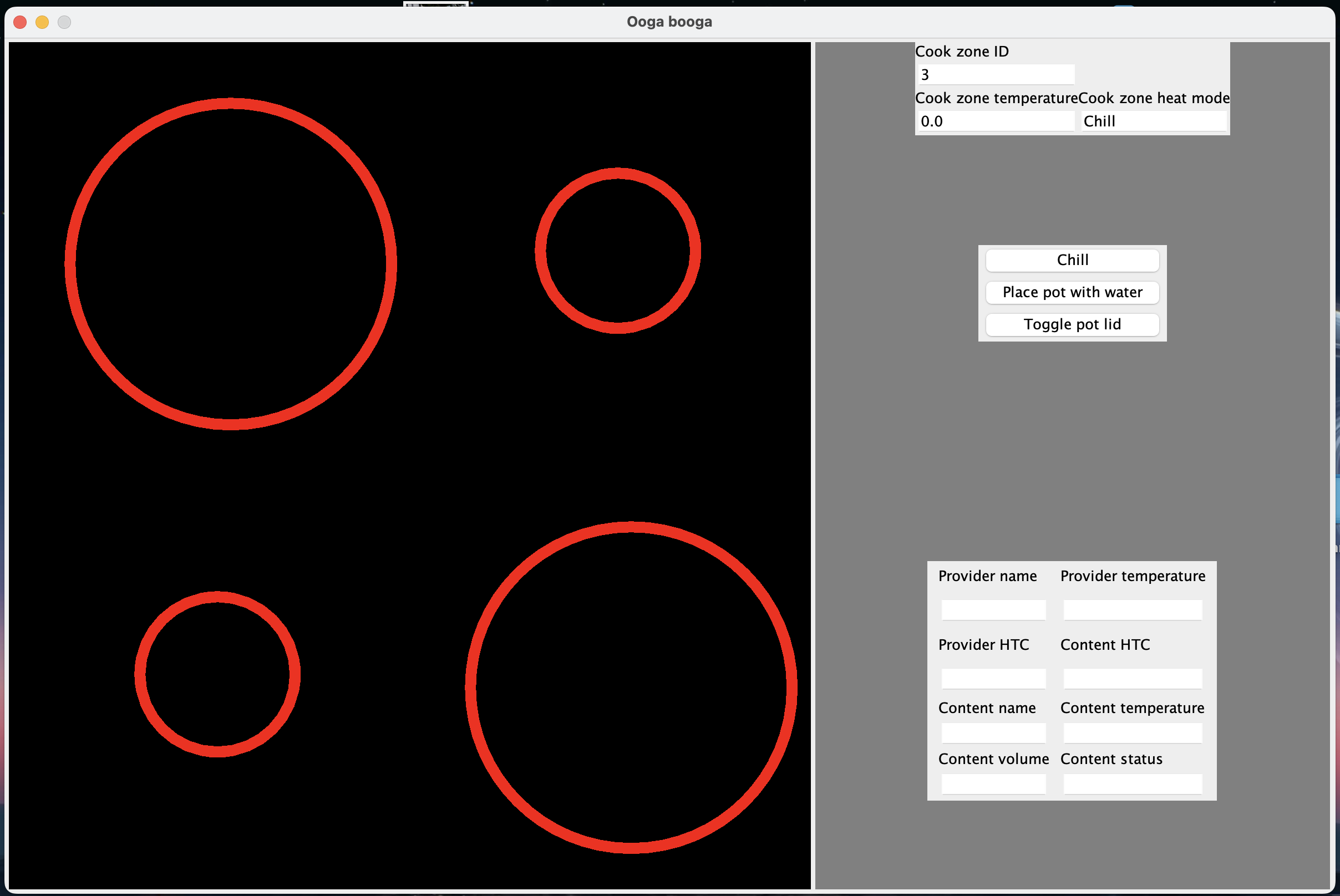


Рисунок 2. Интерфейс приложения

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

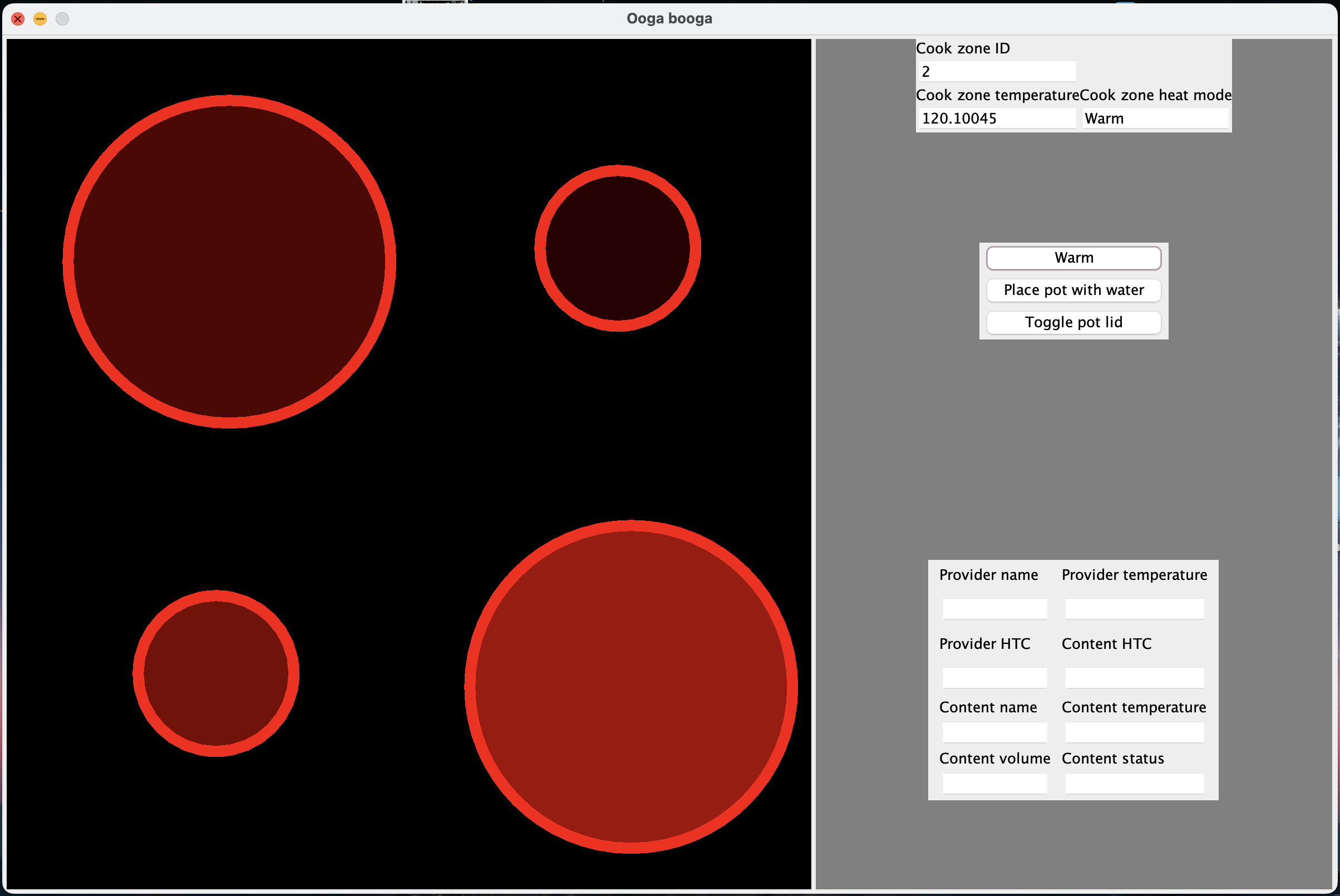


Рисунок 3. Функционирование конфорок плиты

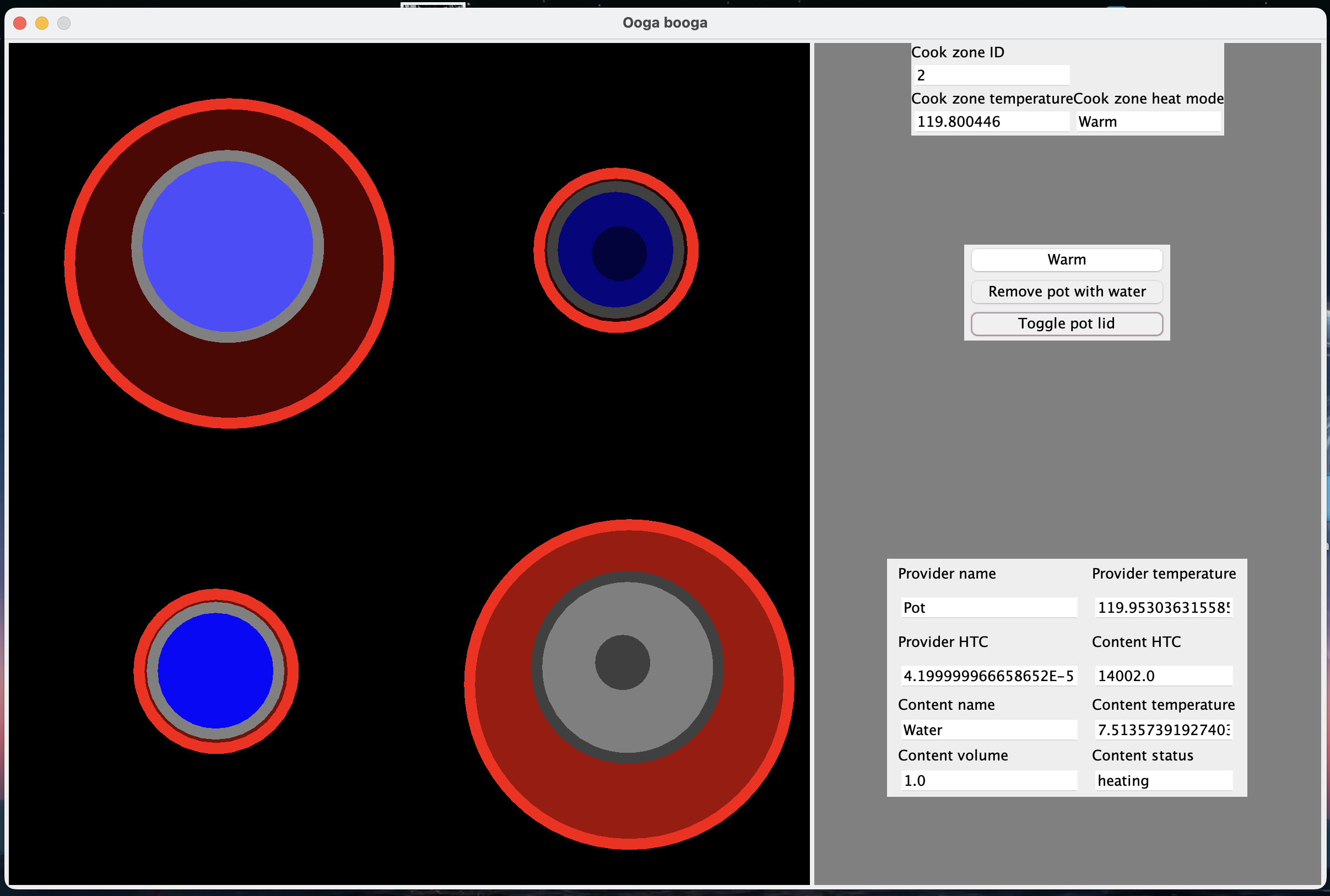


Рисунок 4. Отображение кастрюли на плите (с крышкой и без)

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.

# Устройство приложения изнутри:

1. *Как объект сообщает тепло другому объекту:*

Логика передачи тепла находится в HeatProvider. Эта модель поведения определяет функционал расчета получаемого тепла от другого HeatProvider, а также расчитывает температуру, которая будет сообщена дочернему объекту.



Рисунок 5. Метод включения конфорки плиты.

Поток передачи тепла объекту на конфорке

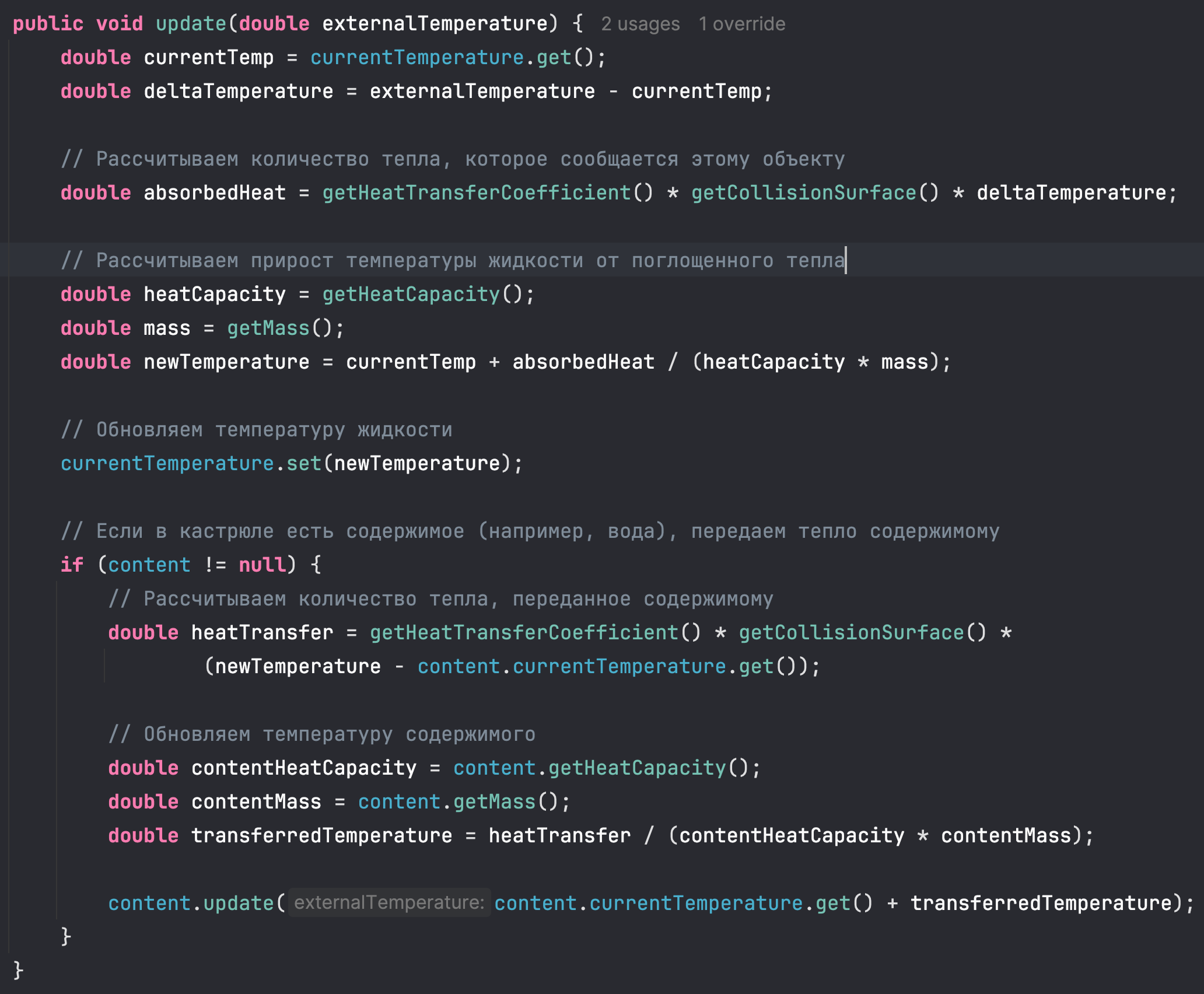
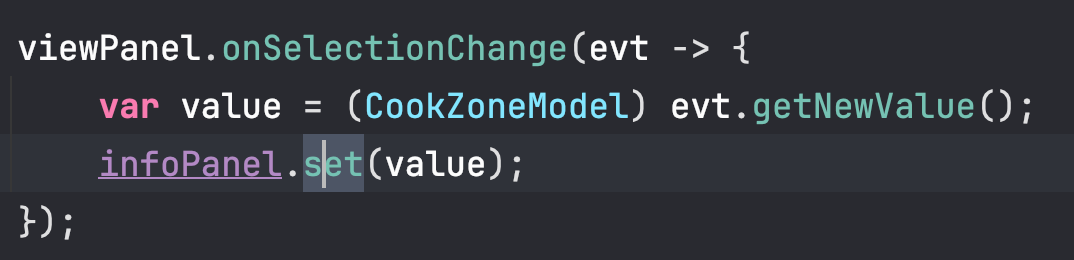


Рисунок 6. Метод для передачи температуры объекту HeatProvider

1. *Как приложение реагирует на действия пользователя:*

Взаимодействие пользователя с системой находится в CookZoneControlPanel. Все подписки на действия находятся там же.  
Чтобы не делать множество уровней вложенности (например, при необходимости среагировать на действие внутри другой подписки), был введен класс SignalBus и SignalListener. Мы подписываемся в MainFrame на клик по конфорке, и говорим слушаетелям, как им нужно отреагировать – можно назвать это правилом поведения.



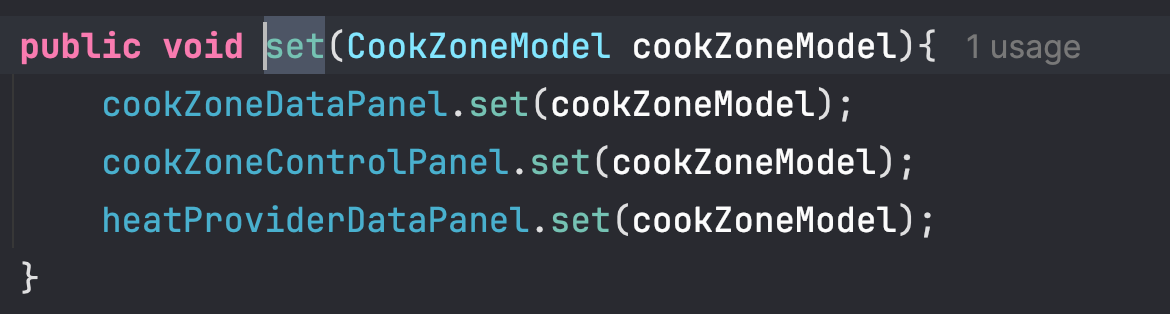


Рисунок 7. Реакция на смену выбранной конфорки в InfoPanel

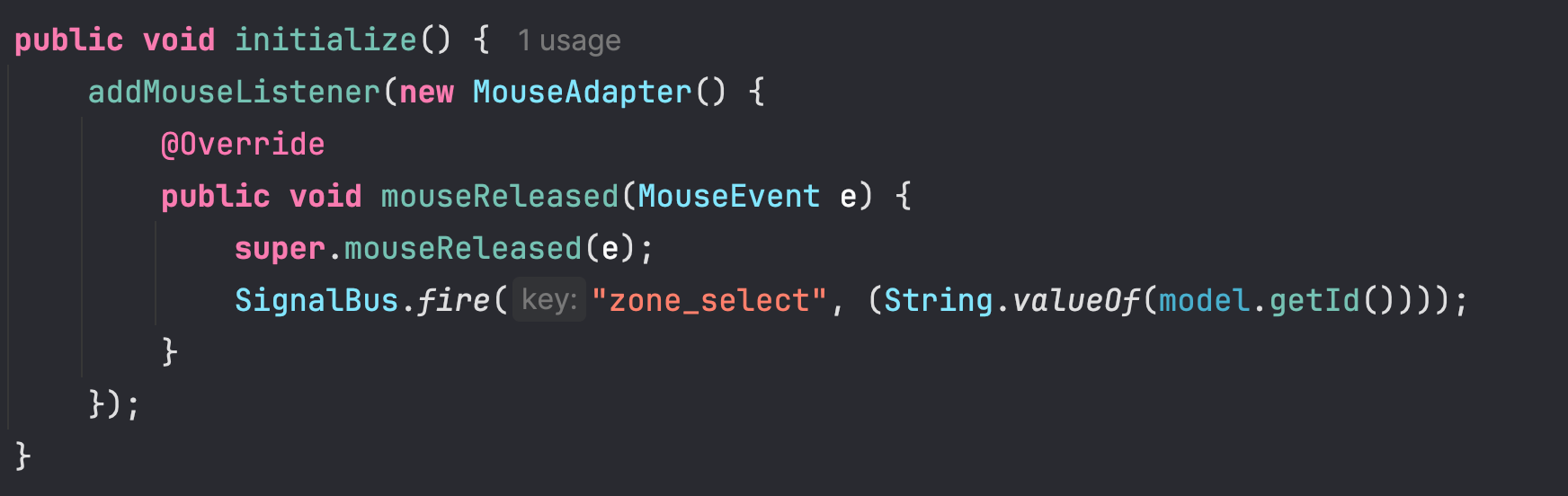


Рисунок 8. Выбор конфорки внутри CookZoneView



Рисунок 9. CooktopViewPanel - создание отображений конфорок и

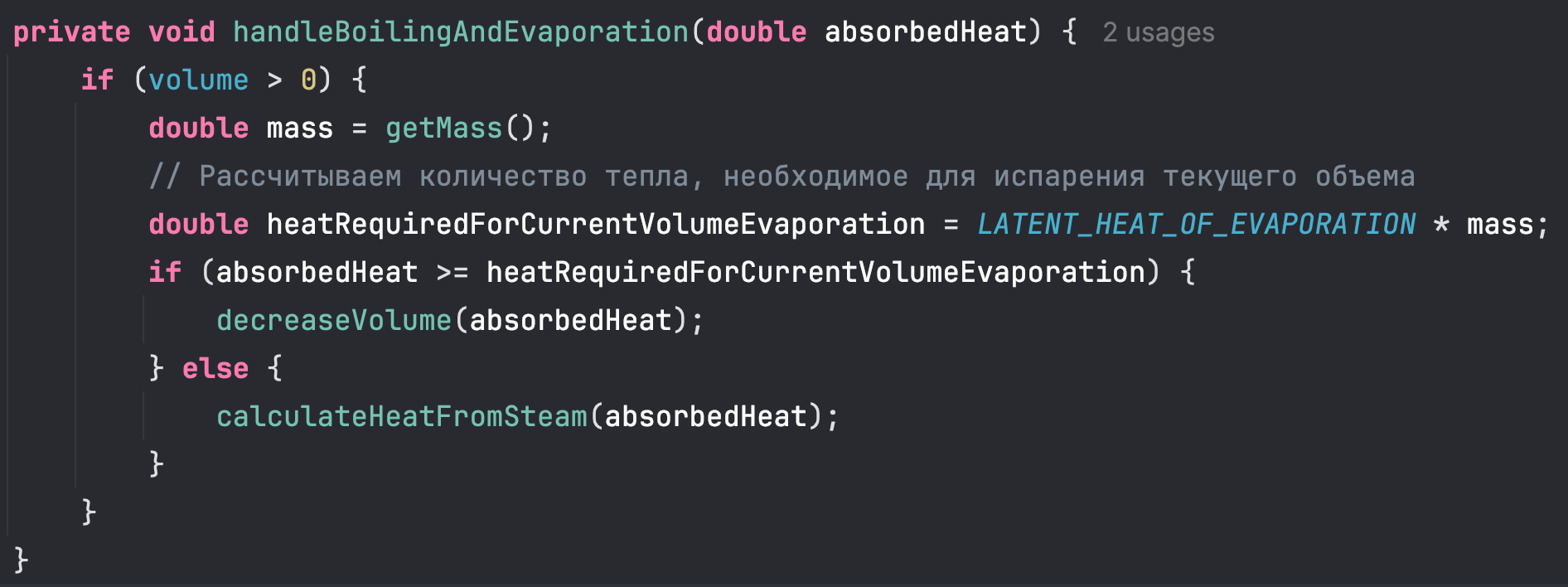
подписка на смену выбранной конфорки

1. *Как происходит переход между состояниями нагрева жидкости:*

⁠ Для представления такой логики, был введен класс StableHeatProvider. Если появится необходимость греть не только воду, можно будет просто поменять аргументы при создании экземпляра класса.



Рисунок 10. Метод для сообщения температуры жидкости. Расчет перехода между фазами нагрева



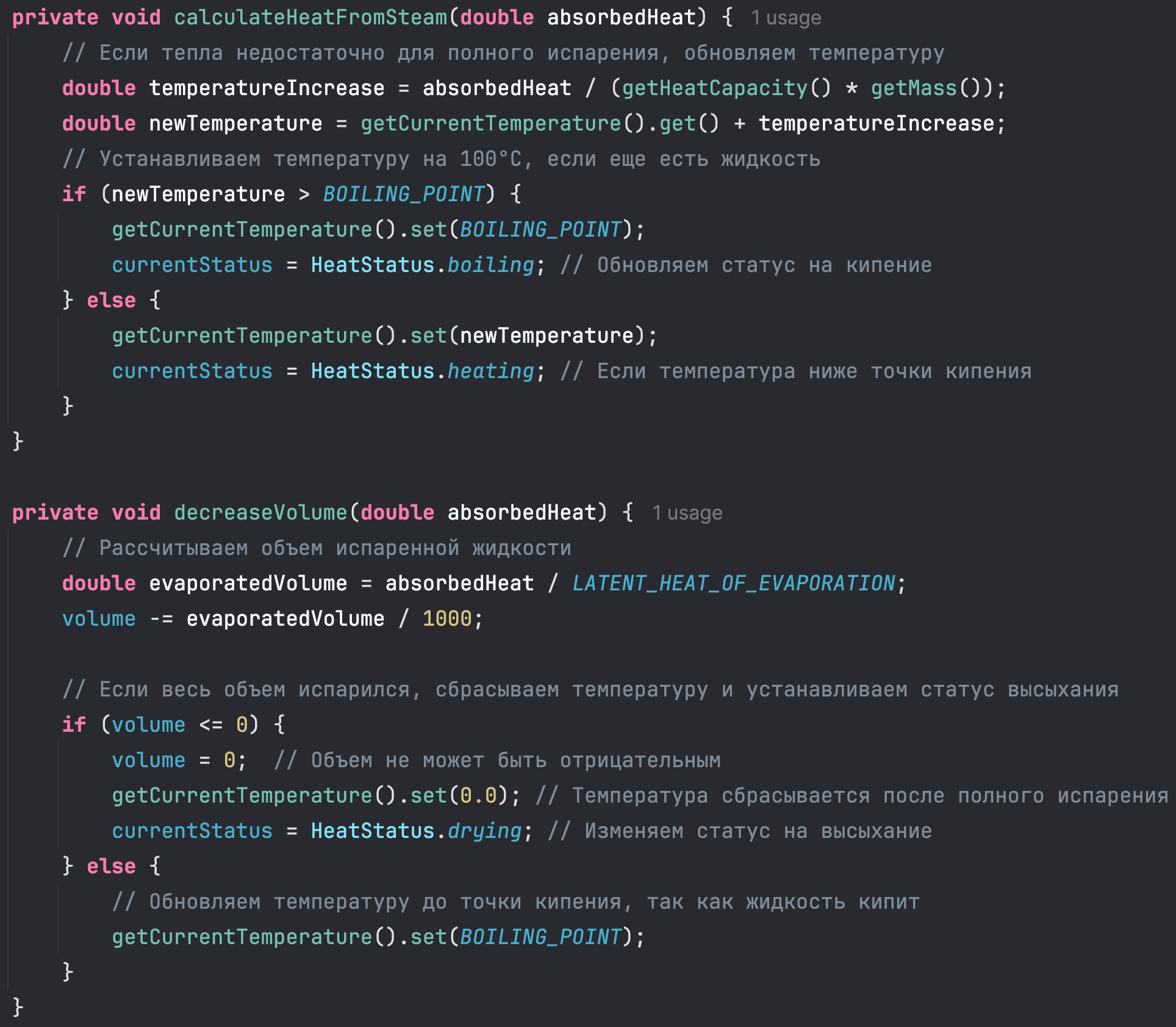


Рисунок 11. Методы для расчета испарения жидкости

# 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