```python
# Event class
class Event:
    def __init__(self, name):
        self.name = name
        self.handlers = []

    def attach(self, handler):
        self.handlers.append(handler)

    def detach(self, handler):
        self.handlers.remove(handler)

    def emit(self, *args, **kwargs):
        for handler in self.handlers:
            handler(self, *args, **kwargs)


# Event handler function
def on_user_created(event, user):
    print(f"User created: {user.name} ({user.email})")

def on_user_updated(event, user):
    print(f"User updated: {user.name} ({user.email})")


# User class that produces events
class User:
    def __init__(self, name, email):
        self.name = name
        self.email = email
        self.created = Event("user_created")
        self.updated = Event("user_updated")

    def create(self):
        self.created.emit(self)

    def update(self):
        self.updated.emit(self)


# Create a user object
user = User("UMEAIMEN", "umeaimen20@gmail.com")

# Attach event handlers
user.created.attach(on_user_created)
user.updated.attach(on_user_updated)

user.create()

user.update()

# Detach an event handler
user.created.detach(on_user_created)

user.create()
# Updating but no handler will react)
user.update()
# No output
```

```
User created: UMEAIMEN (umeaimen20@gmail.com)
User updated: UMEAIMEN (umeaimen20@gmail.com)
User updated: UMEAIMEN (umeaimen20@gmail.com)
```