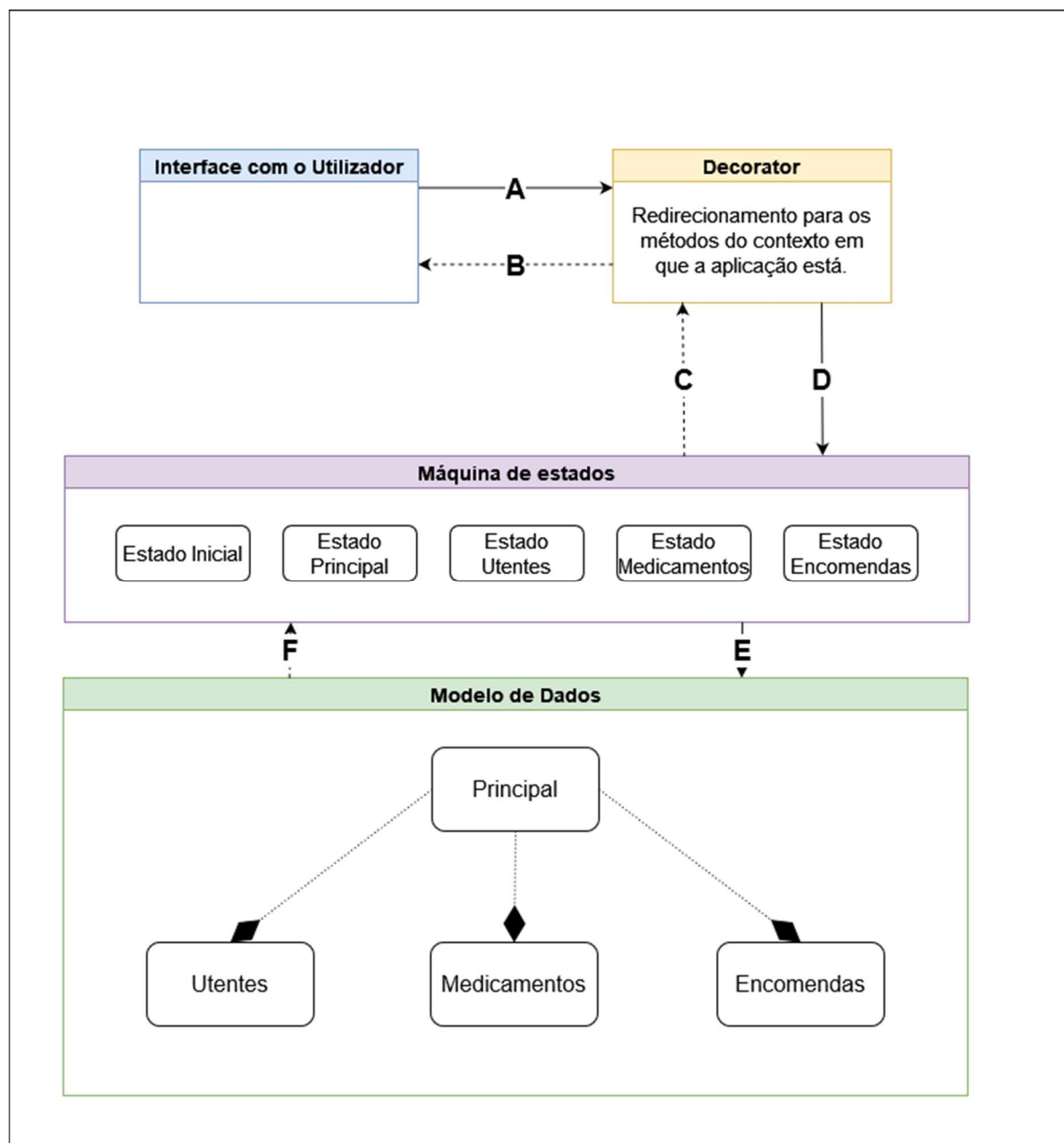


A	Envia dados e Pede Dados
B	Recebe dados/informações
C	Passagem de dados
D	Mediante aquilo que o utilizador pediu em A o decorator redireciona para a função adequada(por exemplo: ver lista de utentes)
E	Acede ao modelo de dados para inserir ou retornar informação
F	Retorno de informação



No Diagrama acima, podemos visualizar a arquitetura do nosso sistema:

- No Modelo de Dados (caixa verde) irá haver uma classe **“Principal”**, que irá conter uma lista de Utentes (classe **“Utentes”**), uma lista de Medicamentos (classe **“Medicamentos”**) e uma lista de Encomendas (classe **“Encomendas”**). É então fácil de perceber, que os dados da aplicação (isto é, dados dos utentes, medicamentos, encomendas) estarão na classe **“Principal”**.
- Temos também uma Máquina de estados (caixa lilás), que vai permitir controlar que ações se podem fazer no contexto em que a aplicação está (por exemplo, é expectável que seja possível fazer encomendas se a aplicação está no “Estado Encomendas”. Isto já não faria sentido se estivesse no “Estado Utentes”).
- Por fim temos o *design pattern*, Decorator (caixa amarela) e a Interface com o utilizador (caixa azul).  
A interface sempre que necessitar consultar informações/dados do modelo de dados ou enviar dados para o modelo de dados, faz através do Decorator.  
O Decorator é que depois trata de redirecionar para as funções adequadas, para devolver a informação ao utilizador ou inserir informação do utilizador no modelo de dados.