

# Software Class Diagram

System classes and their relations



## Vending Machine system



### System Maintenance

#### Product

public:  
+ProductID: enum

private:  
+ logProduct : Log  
+ m\_name : string  
+ m\_value : float  
+ m\_stock : int  
+ m\_id : ProductID

private:  
+ setId(ProductID) : void

public :  
+ getName() : string  
+ getValue() : float  
+ getStock() : int  
+ setName(string) : void  
+ setValue(float) : void  
+ setStock(int) : void  
+ getProductDatabase(vector) : void  
+ setProductDatabase(Product\*vector) : void



### Machine Core

#### VendingMachine

private:  
+ m\_totalMachineCash: float  
+ m\_transactionCash: float  
+ logVendingMachine: Log\*  
+ m\_interface: Interface\*  
+ productDatabase: Product\*  
+ advertisingMessage: Advertising\*

public:  
+ cancelEvent() : void  
+ timerEvent(): void  
+ confirmationEvent(int): void  
+ cashIncrementEvent(float) : void  
+ productSelectionEvent(int) : void  
private:  
+ ST\_Idle(EventData\*) : void  
+ ST\_Develution(EventData\*) : void  
+ ST\_Validation(VendingMachineData\*) : void  
+ ST\_Transaction(VendingMachineData\*) : void  
+ ST\_Deployment(VendingMachineData\*) : void

#### VendingMachineData

public:  
+ cashValue: float  
+ productSelection: int

#### StateMachine

protected:  
+ currentState : unsigned char

private:  
+ m\_maxStates : const unsigned char  
+ m\_eventGenerated : bool  
+ m\_pEventData : EventData\*

protected:  
+ externalEvent(char,EventData\*) : void  
+ internalEvent(char,EventData\*) : void

private:  
+ StateEngine() : void



### Interface

#### Queue

private:  
+ m\_head : Node\*

public:  
+ enqueue(string) : void  
+ dequeue(void) : string  
+ peek(void): string  
+ peekCyclic(void): string

#### Advertising

private:  
+ mainQueue : Queue  
+ removedQueue; Queue

public:  
+ getAdvertising() : string  
+ getAdvertisingDatabase(void) : void

#### Node

private:  
+ m\_nodeData : int  
+ m\_nextNode : Node\*

public:  
+ getNodeData() : int  
+ getNextNode() : Node\*  
+ setNodeData(int) : void  
+ setNextNode(Node\*) : void

#### DebugInterface

+ m\_productSelected: int

public:  
+getSystemInput(SystemData\*) : void  
+setSystemOutput(SystemData\*) : void

+getUserInput(UserData\*) : void  
+setUserOutput(UserData\*) : void

+printAdvertising(AdvertisingData\*) : void  
+insertAdvertising(AdvertisingData\*) : void

#### Interface

public:  
+getSystemInput(SystemData\*) : virtual void  
+setSystemOutput(SystemData\*) : virtual void

+getUserInput(UserData\*) : virtual void  
+setUserOutput(UserData\*) : virtual void

+printAdvertising(AdvertisingData\*) : virtual void  
+insertAdvertising(AdvertisingData\*) : virtual void



### Debug System

#### Log

public:  
+ Level : enum

private:  
m\_logLevel : Level  
m\_scope : string  
logFile: ofstream  
logFileLocation: string

public:  
+ setLevel(Level) : void  
+ error(string) : void  
+ warn(string) : void  
+ info(string) : void  
+ debug(string) : void