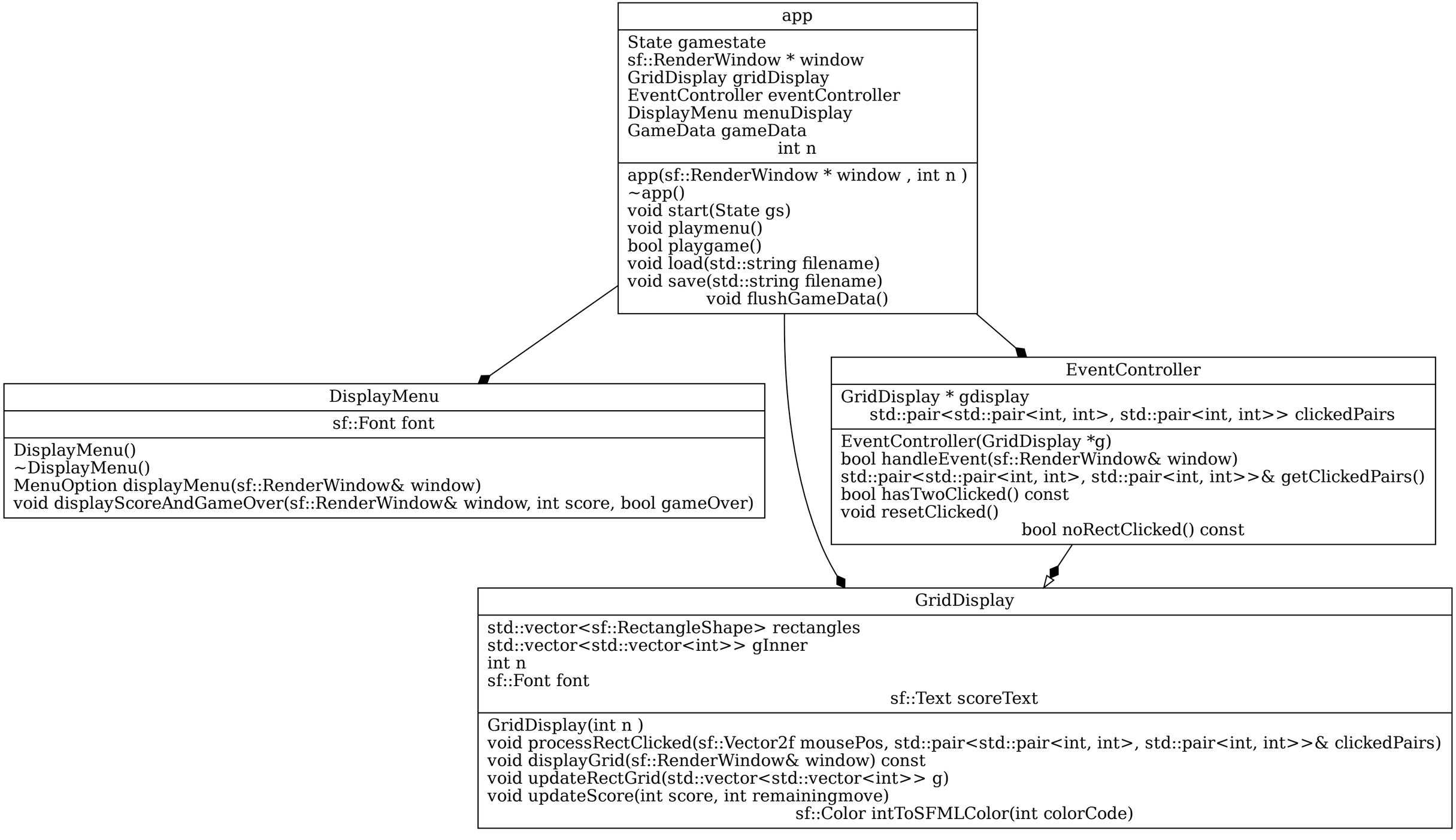
app
State gamestate
sf::RenderWindow * window
GridDisplay gridDisplay
EventController eventController
DisplayMenu menuDisplay
GameData gameData
int n
app(sf::RenderWindow * window , int n )
~app()
void start(State gs)
void playmenu()
bool playgame()
void load(std::string filename)
void save(std::string filename)
void flushGameData()
DisplayMenu
sf::Font font
DisplayMenu()
~DisplayMenu()
MenuOption displayMenu(sf::RenderWindow& window)
void displayScoreAndGameOver(sf::RenderWindow& window, int score, bool gameOver)
EventController
GridDisplay * gdisplay
std::pair<std::pair<int, int>, std::pair<int, int>> clickedPairs
EventController(GridDisplay *g)
bool handleEvent(sf::RenderWindow& window)
std::pair<std::pair<int, int>, std::pair<int, int>>& getClickedPairs()
bool hasTwoClicked() const
void resetClicked()
bool noRectClicked() const
GridDisplay
std::vector<sf::RectangleShape> rectangles
std::vector<std::vector<int>> gInner
int n
sf::Font font
sf::Text scoreText
GridDisplay(int n )
void processRectClicked(sf::Vector2f mousePos, std::pair<std::pair<int, int>, std::pair<int, int>>& clickedPairs)
void displayGrid(sf::RenderWindow& window) const
void updateRectGrid(std::vector<std::vector<int>> g)
void updateScore(int score, int remainingmove)
sf::Color intToSFMLColor(int colorCode)