Scientific Python Proposition

Kalcsevszki Regina, Kiss Dorottya Tünde, Márton Adrienn

Purpose (what the program is to achieve)

In the Wallbreaker Game the user plays with a bat to control a ball to tear down the wall. If the ball hits a brick, usually the brick will disappear, and the ball will bounce back. Since it is a game the main goal is to entertain the user.
The player will complete a level, when all the bricks (without the unbreakable) are broken.
After the user goes level to level, it will be harder and harder. The bricks wouldn't break for the first time, and to complete the level the player has to be more accurate. There can be unbreakable bricks too.
The player has limited life and if all balls fall down, the player loses a life. The ball only leaves the screen in the bottom, the ball will bounce back from the other three bound.
When a brick falls apart a bonus can fall. The user can decide to catch it or just leave it to fall down. The bonuses could be gifts, which make the game easier, or could be penalties, which make the game harder. Just a few bonuses: fire ball, bigger/smaller bat, the ball speed increase/decrease, bigger/smaller ball etc. Some of the bonuses will be temporary, so those will disappear after time. If all of the balls fall down, the new ball has no bonuses, but the bat can keep its bonuses.

Functional specification (describes what has to be implemented)

The game has to start with a Menu. There, they can choose to start a New Game, to Load an older save, or read a How to, which describes the game, and how to play it.
Every first time, when a new game starts, the player can choose to play without the bonuses.
The game has to autosave after a level completed.
The ball has to move according to the physics law, but it's velocity doesn't change (just in case a bonus affects it).
The ball couldn't leave the gamespace, because it bounces back from the upper, left and right bound of the screen, and if it falls down, the player loses a life, and the ball has to disappear.
If the user gets a new ball, all the ball's bonuses have to disappear.
The user can play a completed level again, but can't play unlocked levels, which are after the actual level.

Systems architecture (describes how the functions will be realized)

Main classes: Ball (and it's child classes), Bat (and it's child classes), Brick (and it's child classes), Bonus, GUI, Menu, Level, Game, Save.