**Shoot the Goomba’s – Design Document**

*List of database tables and fields*

High Score table in SQLlite with two fields:

* name: TEXT contains the name of the user
* points: INT contains the number of points the user has scored

SharedPreferences in a XML file “*goombasPrefs*”

* name: String. Contains the name of the user
* music: boolean. True if the user has chosen the music to be on.

*List of classes and methods*

MainActivity: Controls the home page.

* public void startGame(View view): Called when the Start button in the view is clicked. Starts an intent to the gameActivity class with the extra value it’s a new game.
* public void highScores(View view): Called when the high Scores button in the view is clicked. Starts an intent to the scoreActivity class.

gameActivity: Controls the Goomba game

gamePlay: Controls the view for the Goomba game.

* field variables
  + private final Context myContext;
  + private SharedPreferences prefs
  + private int points : number of points scored by in the game
  + private int numberOfBullets : number of bullets left in the game
  + private boolean music : true if music is on
  + private MediaPlayer mpReload : sound for reloading bullets
  + private MediaPlayer mpGoomba : sound for shooting a goomba
  + private MediaPlayer mpEmpty : sound for shooting without bullets
  + private MediaPlayer mpBackground : background music
* constructor
  + this.myContext = context;
  + this.prefs = preferences;
  + this.points = 0 : set scored points to zero
  + this.numberOfBullets = 10 : set number of bullets to ten
  + this.music = prefs.getBoolean("music", true) : get value of music from SharedPreferences
  + mpReload = MediaPlayer.create(myContext, R.raw.gun\_cocking\_01);
  + mpGoomba = MediaPlayer.create(myContext, R.raw.smw\_stomp);
  + mpEmpty = MediaPlayer.create(myContext, R.raw.gun\_empty);
  + mpBackground = MediaPlayer.create(myContext, R.raw.super\_mario\_bros);
  + mpBackground.setVolume(0.0f, 0.8f);
  + mpBackground.start();
* Methods
  + switchMusic : change the value of the music from on/off
  + playEmpty: play the sound of an empty bullet
  + playShotGoomba: play the sound of a shot goomba
  + playReload: play the sound of reloading a bullet
  + muteBackground: set the volume of the background music to zero
  + stopMusic: stop the background music
  + updatePoints: update the number of scored points
  + updateBullets: Update the number of used bullets
  + addBullets: add parameters for a single bullet
  + setReloadButton: set parameters for the reload button
  + getNumberOfBullets :
  + getMusicValue
  + getPoints

Goomba: class for a goomba, extends the button class

* Field Variables
  + Int y : Starting height of the goomba
  + Boolean fromLeft : true if the goomba starts from left
  + Int value: value of the goomba
* Constructor
* Methods