**Unit Test Skeleton**

public class Test{

**Game Test Cases**

public void gameAnimations(){

//tests the game animations

}

public void testQuestionGeneration(){

//tests that game generates random question with 4 answer options. One of which must be the correct answer to the question

}

public void testKeyAnswers(){

//tests that game correctly responses to answers submitted through the keyboard (up, down, left, right keys)

}

public void testIncorrectAnswer(){

//tests that game adds time penalty for incorrect answers

}

public void testScore(){

//tests that game records time to complete course

}

public void testScoreboard(){

//tests that game saves tops scores of students

}

public void testGameStats(){

//test that all game stats are saved properly in the database

}

**Paypal Test Cases**

public void testPayLogin(){

//Tests that the PayPal login works

}

public void testPayGuest(){

//Tests PayPal guest checkout

}

public void testPayMember(){

//Tests PayPal member checkout

}

**Twitter Test Cases**

public void testTwitterLogin(){

//Tests twitter login,

}

public void testTwitterLink(){

// Tests link success/failure

}

public void testTwitterPosting(){

//Tests posting ability (Pictures, comments, retweets)

}

**Facebook Test Units**

public void testFBLogin(){

//Tests Facebook login,

}

public void testFBLink(){

// Tests link success/failure

}

public void testFBPosting(){

//Tests posting ability (Pictures, videos, comments, shares)

}

**Instagram Test Unit**

public void testInstaLogin(){

//Tests Insta login,

}

public void testInstaLink(){

// Tests link success/failure

}

public void testInstaPosting(){

//Tests posting ability (Pictures, videos, comments, shares)

}

**User Login Database Test Unit**

public void testParentInfo(){

//test that all inputted Parent info is saved properly in the database

}

public void testStudentInfo(){

//test that all inputted Student info is saved properly in the database

}

public void testOfficialInfo(){

//test that all inputted Student info is saved properly in the database

}

**Registration Test Unit**

public void testStudentRegistration(){

//tests that students can register

}

public void testParentRegistration(){

///tests that parents can register

}

**Submission Test Unit**

public void testPhoto(){

//tests that photos can be submitted

}

public void testReview(){

//tests reviews can be submitted

}

public void testParkDescription(){

//tests that park descriptions can be submitted

}

**Report Test Unit**

public void testOfficialReport(){

//tests the automatic generation of an official report after the game is played, which includes each students high score

}

public void testParentReports(){

//tests the automatic generation of an official report after the game is played, which includes each students high score

}

}

**Integration Testing Skeleton**

public class Test{

public void testGame(){

//Checks and combines all the Game Test Cases

}

public void testDonations(){

//Checks and combines all the PayPal test cases

}

public void testSocialMedia(){

//Combines and checks Twitter, Instagram and Facebook test cases

}

public void testDatabase(){

//Combines and checks User Login Databases

}

public void testRegistration(){

//Combines and checks Registration Test Unit

}

public void testSubmissions(){

//Combines and checks Submission Test Units

}

public void testReports(){

//Combines and checks Report Test Unit

}

}

**Regression Test Procedure**

When changes are integrated into the code we will follow the following procedure to ensure the code does not create a bug. We will prioritize certain tests due to their importance and to save time and money. If any of these tests present a bug in the system, stop and begin a top down approach to find it, for example, try the integration tests, and if they do not expose the bug, then go all the way down to unit tests to isolate the issue.

1. Re-run testDonations()
2. Re-run testDatabase()
3. Re-run testGame()
4. Re-run testRegistration()

These four pieces are the most critical to the system success.

**System Test Procedure**

Our system test procedure can be completed in levels; unit testing, integration testing, and system testing. Our first order of business will be to follow the unit test skeleton outlined above to ensure that all small functionalities are working. We will then test all the bigger pieces by using integration testing; we will combine all our units and test each class and ensure that we are getting correct outcomes. Our final step will be to test the whole system together. We will use all the classes outlined and ensure they’re working as well. If there are changes or errors at any steps we can refer to the prioritization discussed in the regression testing procedure to decide how we will find the bug.