Walk through our GameCentre

1. What is your unit test coverage?
2. For our unittest, we basically covered most of the logical parts of our applications, including the control parts for models in MVC pattern and interaction parts with other classes.
3. What are the most important classes in your program?
4. The most important classes are BoardManager classes in Sliding Tiles and Memory Matching as well as UserManager class and ScoreManager class. Since the BoardManager classes in the two games are in charge of most interaction functionality, such as manipulating all the tiles or cards on the board. UserManager are closely related to all the dates from the users, which is the most fundamental step of playing the games. ScoreManager class relates to the other important factor – the scores.
5. What design patterns did you use? What problems do each of them solve?
6. In terms of design patterns, we have used MVC, Factory and Observer. Observer makes our autosave concise and stable, in addition, it updates the view on our phone screen. MVC is what the users will get into for the first time, it handles the view on screen and the data that will be changed along with the change of views, thanks to the help of controllers. Factory Pattern is the way we use to build our game activity and scoreboard activity, giving a steady base for the whole game.
7. How did you design your scoreboard? Where are high scores stored? How do they get displayed?
8. Our leaderboard is in time-major order,so the player who took less time to finish the game will gain a higher position on the leaderboard. The hign scores are stored in a HashMap based on username as keys and in long type (since the time is calculated on milliseconds for accuracy issues). The highest 5 scores are taken to show in a discrete activity (ScoreBoardActivity).