FamiOwl

S-Team

S-Team: Qichao Lin, Xinyu Meng, Yuchen Wei, Wendi Huang, Chengyu Liang

System Requirements:

FAM-001 The system shall allow parents users to set the time limit of game time for each of their children on the parent portal.

FAM-002 Parents can set time limits for each game separately on the parent portal.

FAM-003 Once a child reached the time limit their parents set for this kid id, the game would automatically exit.

FAM-004 Parents can see how much time in total their children have played for each game based on the kid id on the parent portal.

FAM-005 Children can download, install, and play games in one tap.

FAM-006 Each child can use their own id and has their own profile.

FAM-007 Each child has their own game library based on their id.

FAM-008 Each game has its own game profile page.

FAM-009 The child client can record game time for each child based on their id.

FAM-010 Parents need an account to use the parent portal and the child client.

FAM-011 The modification of any time limit rule would apply the next time the corresponding game starts.

FAM-012 The data and profiles of children shall persist based on the data in the database when the parent sign-in to the child client on a different machine.

FAM-013 Games in the library shall have a big square icon with their names under to make them easy to select.

FAM-014 Games shall have tags for categorizing.

FAM-015 Children and parents can search games based on tags.

FAM-016 Children and parents can search games based on names.

FAM-017 Children and parents can hit “Like” to a game.

FAM-018 The system shall display the number of “Likes” a game has.

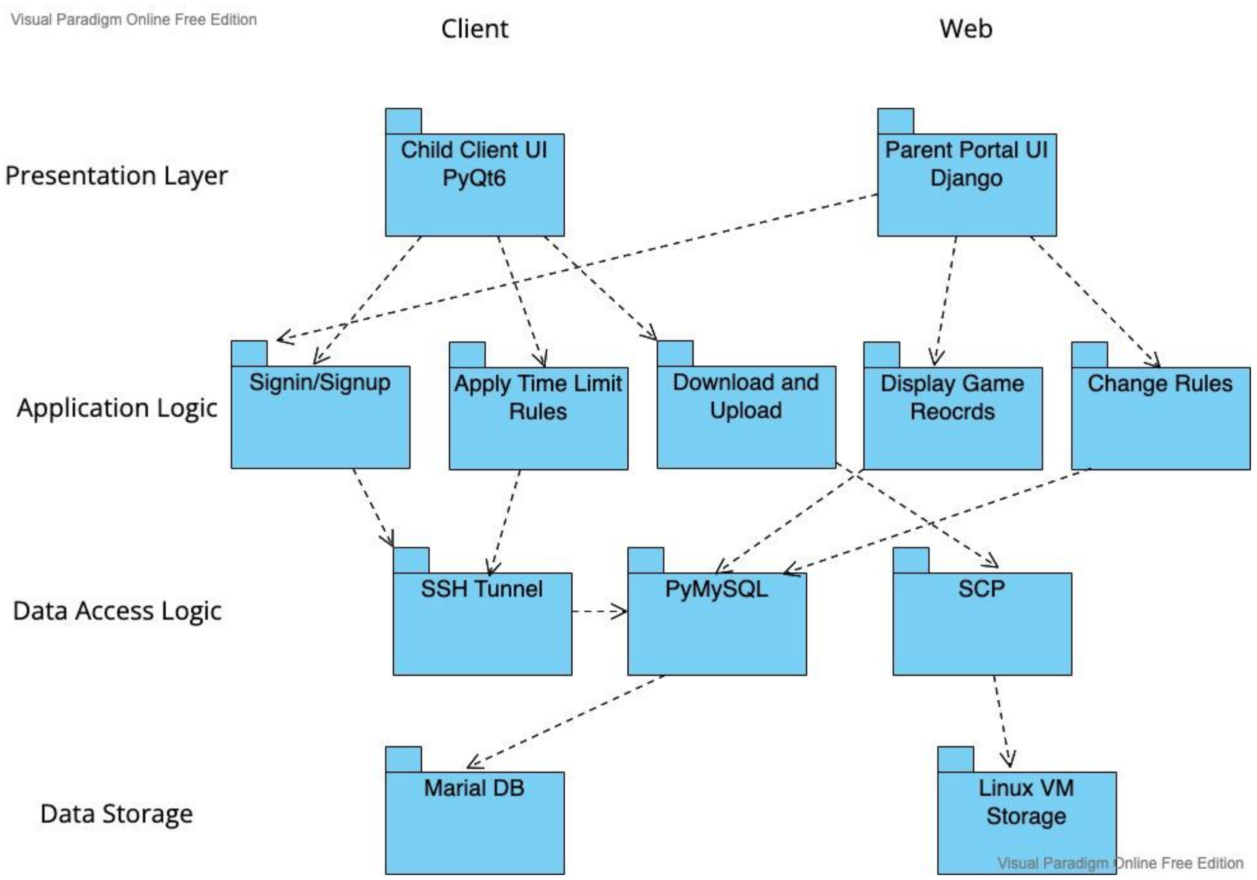
FAM-019 The child client shall be able to let parents sign-up.

FAM-020 The parent portal shall be able to let parents sign-up.

FAM-021 Elements, icons, UIs shall resize and adapt automatically when the parent is accessing the parent portal on their mobile devices.

FAM-022 Parents can also set an overall time limit for any game.

Software Architecture



Initial Test Design:

We are planning to use pytest and django built-in test modules to test our project.

For our web server, we will use django.test to simulate POST and GET functions, we will go through each of the view functions to test if each urls are working correctly for integration testing.

For the child client, we will test the qt functions thoroughly and then we can also do systematic testing to connect the two clients.