TDA367: Object-oriented programming project
Chalmers University of Technology

Grey Matter

System Design Document

and I OOPP:
ALINE EIKELAND
HUGO GANELIUS
VIKTOR JOHANNESSON
EMIL LUNDQVIST
FELIX OLIV

1 Introduction

This system design document intends to introduce the reader to the brain test mini game android application that is Grey Matter. The document will develop on design choices and process, system structure and qualities as well as the finished product.

1.1 Definitions, acronyms, and abbreviations

Gamification - applying gameplay principles on non-gaming activity to increase attractiveness
JSON - JavaScript Object Notification, easily readible file format for data storage
JSON-server - JSON module to easily mockup servers without having to program server part.
GSON - Googles JSON server service.
MVC - Model View Controller pattern.
MVVC - Model View ViewModel pattern.
Normal distribution - Natural probability calculation, looks like a bell curve.
Standalone application - A application that runs locally on the device and does not require anything else to be functional.

2 System architecture

GrayMatter is a mostly standalone application, and all the logic is therefore built into the app. However to be able to compare your score with other players, the app needs a internet connection to download other players data which is located on a server.

Grey Matter application contains all necessary logic and GUI. The social interaction and leaderboard features however requires the need of a database and server. If the player experiences infrequent internet connection, the offline features can still be used although the results will not be updated until the player retains internet access.

2.1 Flow of the application

When the application is started the method on-Create in the Main-activity class is called. This method will instantiate the main-page XML file which displays the main-page to the,

3 System design

3.1 Modular

3.1.1 MVC-pattern

3.1.2 MVVC-pattern

3.2 One purpose

Assigning a single responsibility to a package or class goes hand in hand with the last subsection on modularity.

3.2.1 Data Mapper

Data Mapper is not a traditional Object oriented design pattern. Although it is based around the principle

3.3 Foolproof

3.4 Every edge case noticed

3.5 High abstraction

4 Persistent data management

Grey Matter uses a JSON server for storing user data and game records.

5 Quality

5.1 Testing

The application is tested with jUnit tests and Travis automatic build testing.The project can be viewed continously at this github repo.

5.2 Issues

The code has mostly healthy dependecies.The social package has a file, PlayerMapper, which needs to be seperated into smaller areas of responsibilities. It is too long and it not following the DataMapper pattern as intended.

5.2.1 PMD results

PMD test found problems mostly related to small fixes as failure to remove unused code and imports, extra brackets of different kinds and misadaption to Java conventions, such as capital letters in the beginning of a package name.
It did however also highlight following mistakes in the code:

- Typing to implementation class instead of interface - this limits flexibility.
- Using unnecessary local variables where one could immediately return result - this clutters the code and uses additional storage.

For full PMD analysis see appendix.

5.3 Access control and security

Grey Matter uses personal accounts for storing results and social interaction.

6 Appendix

7 References

Gradle documentation
Gson documentation