|  |  |  |  |
| --- | --- | --- | --- |
| **Team Number** | 05 | **Section** | 03 |
| **Team Members** | 1. Brandon Truong 2. Jason Khy 3. Phong Nguyen | | |
| **Software Name** | Ruminate Self-Study© | | |

## Introduction

* 1. **Objective** [10 Points]

This document will serve as a technical specification to determine and document the components to be implemented within the software. We intend for the readers (our developers) to extract a general understanding of the structure of our software. This document includes items that require pre-existing knowledge of:

1. Java Programming Language
2. UML Diagrams
   1. **References** [5 Points]

Adobe XD

Violet UML

Google Docs

Professor Yazdankhah’s Lecture Slides

* 1. **Acronyms, and Abbreviations** [5 Points]

|  |  |
| --- | --- |
| Adobe <XD> | Adobe <Experience Design> Software to create mockups. |
| Violet <UML> | Violet <Unified Modeling Language> Software to graphically plan out the design phase. |
| <OS> | <Operating System> System software that manages the computer components and runs applications. |

## Software Overview

* 1. **Problem Statement** [10 Points]

We believe that it is essential to be able to self-study in an effective manner. To do so would require proper material distribution and pacing, which is difficult to do alone. To solve this problem, we will create a software that not only compiles your study phrases, but takes into account their difficulty and rearranges your study time to review efficiently.

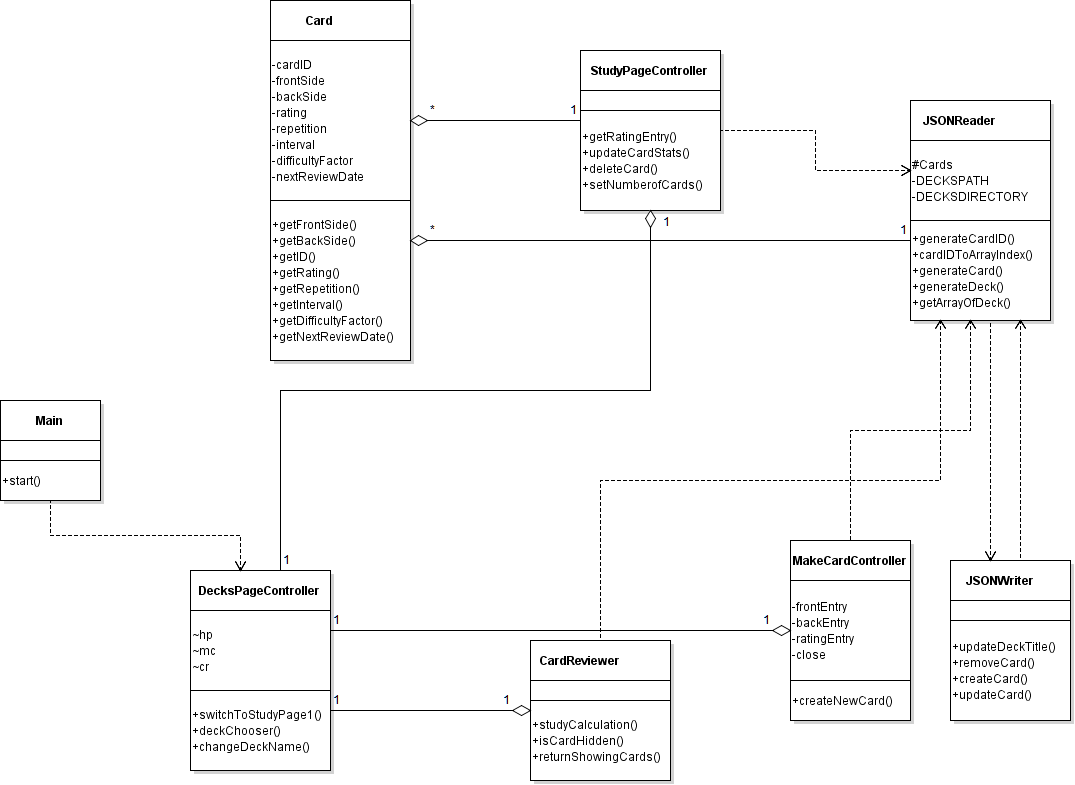
* 1. **Scope** [5 Points]

The intended users for Ruminate Self-Study are people who have to memorize key terms or concepts through spaced repetition.

* 1. **Technological Requirements** [10 Points]
* Technologies
  + Java
  + JavaFX
* Technological Requirements
  + Desktop OS (No mobile support)
    - ie. Windows, Linux, MacOS
  + Latest Java Install

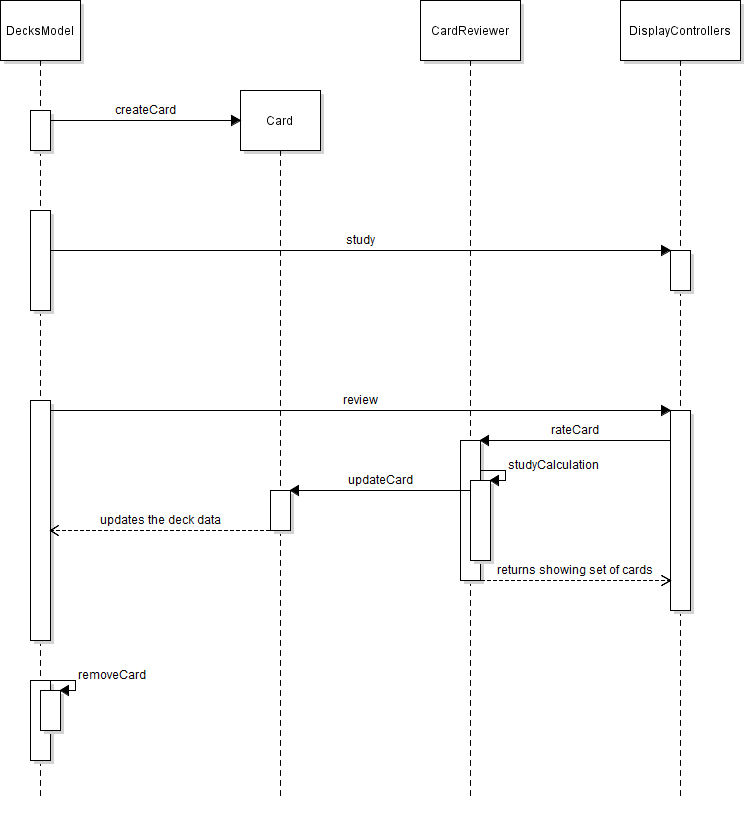
## Detailed Design

* 1. **UML Class Diagrams** [20 Points]



* 1. **UML Sequence and/or State Diagrams** [15 Points]

|  |  |
| --- | --- |
| User (actor) Action | System (actor) Response |
| 1. User creates card in deck | 2. System creates card |
| 3. User study/review  User rates difficulty | 4. System sets difficulty  System calculates time to review  System sets time to review cards  System return card with updated info |



* 1. **User Interface** [15 Points]

