Virtual Classroom (Vlass)

Detailed Design

COP 4331C, Fall, 2015

## **Modification History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Who | Comment |
| V0.0 | 10/15/2015 | J. Casserino | Initial Draft |
| V0.1 | 10/27/2015 | C. Armstrong  J. Bender  J. Casserino  M. Friedman | Rough Draft  - Added Design Issues  - Added Class Diagram  - Added Sequence Diagram  -Added Trace Requirements |

Team Name: Group 26

Team Members:

Joseph Bender [jbender94@knights.ucf.edu](mailto:jbender94@knights.ucf.edu)

Joshua Casserino [Joshua.casserino@knights.ucf.edu](mailto:Joshua.casserino@knights.ucf.edu)

Chad Armstrong [chad.armstro@knights.ucf.edu](mailto:chad.armstro@knights.ucf.edu)

Miles Friedman [milesfriedmanfl@gmail.com](mailto:milesfriedmanfl@gmail.com)

Contents of this Document

Design Issues.........................................................................................................................3

Detailed Design Information.................................................................................................5

Trace of Requirements to Design..........................................................................................7

**Design Issues**

**Reliability**

In order to provide a reliable application, a reliable back-end architecture is the most critical component for reliability. Many third party data web services were analyzed until a reliable candidate, Parse, was selected to integrate with the application.

**Reusability**

The swift code in the iOS app is written in the form of MVC and class structures. This structuring of the code allows for operations and attributes to be shared across view controller classes. This feature of the design allows the developers to simplify the design and shorten development time, from not having to write as much code.

**Maintainability**

Implementing the admin perspective into the application is a critical feature to ensure maintainability. The admin account is hard coded into the database with specified strict credentials. This account has all of the functionalities and abilities of normal app users, and also contains extra functionalities to provide additional services to the normal users, and revert system errors that may occur.

**Testability**

The app is to be heavily unit tested throughout development to ensure all features are functional before the developers move forward in the design. The results are verified with the expected outcome. The app is also tested as a whole on a physical device. It is written to a physical device to ensure that no discrepancies occur between the simulator and the physical device.

**Performance**

The performance of the Vlass application is a critical feature of the design. In order to achieve a maximum performance, all query operations to the Parse data service must be made as efficiently as possible. Querying of redundant data or non-essential data is avoided at any occurrence. Also, the internal algorithms of the application to verify data and receive data are all to be within the runtime of O(n2).

**Portability**

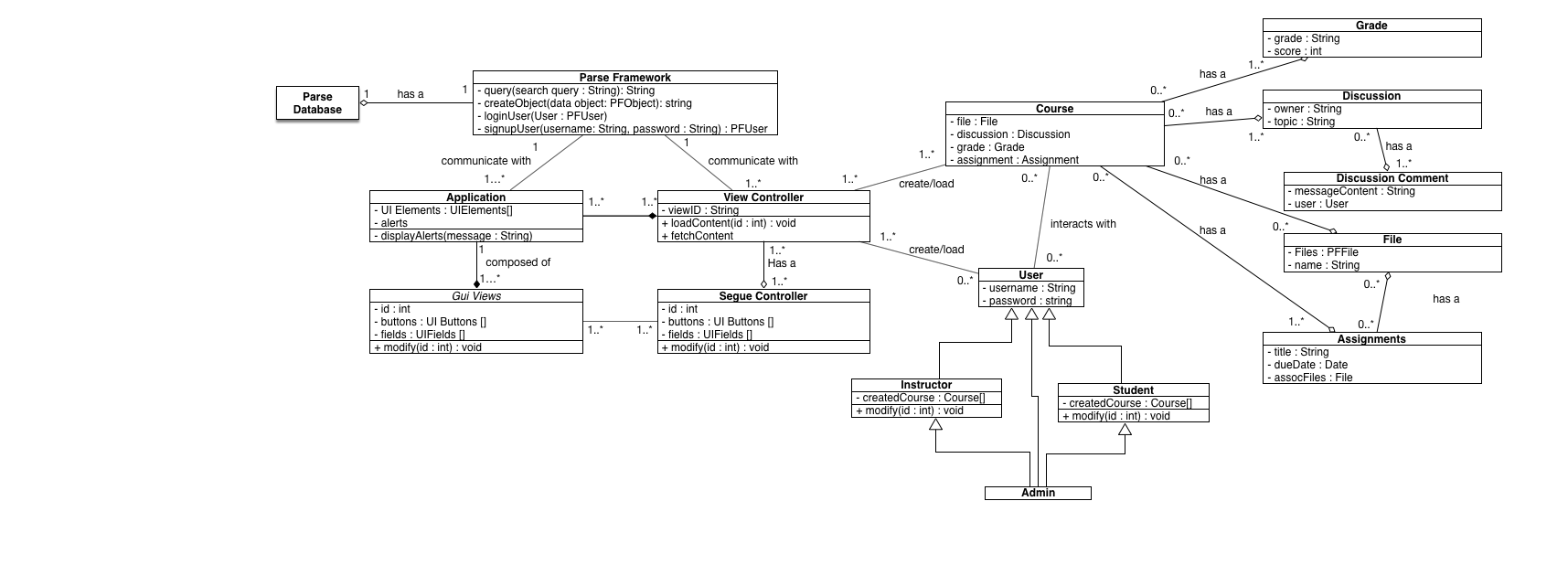
If the application were taken to market on the Apple iTunes© Store, it would provide a great sense of portability in that the UI design has been setup to work with a variety of iPhone and iPad screen sizes. This requires that all UI elements in each perspective are set with the correct constraints to accommodate any resolution.

**Security**

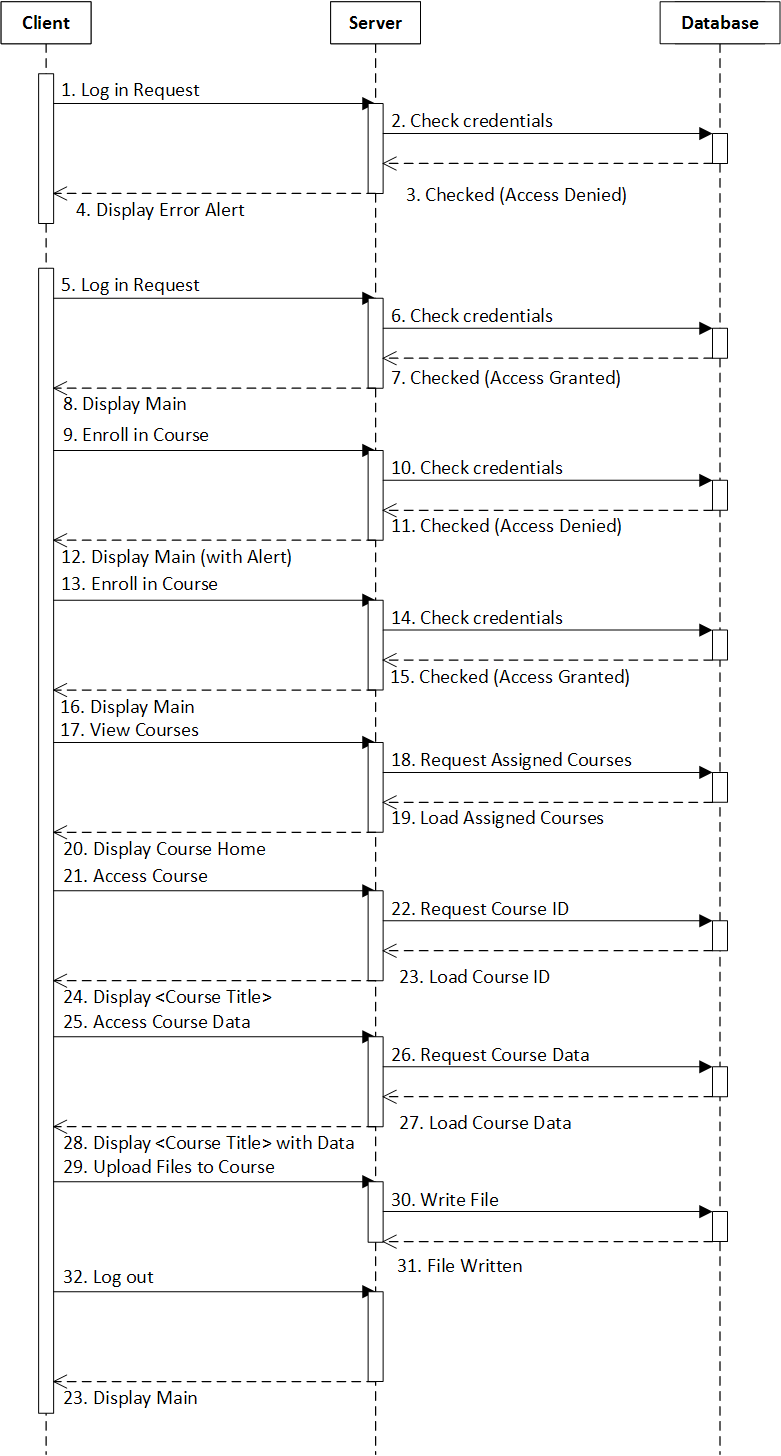
The security of the Vlass application is evident in the user credential enforcement. The password criteria were determined from field standards for validation. The credentials are thoroughly enforced in the swift code of the application. The transfer of data to the Parse data service is also encrypted. This is another feature that led to the decision to select Parse.

**Detailed Design Information**

**Class Diagram**



**Sequence Diagram**



**Trace of Requirements to Design**

|  |  |
| --- | --- |
| **Design Component** | **Requirement Number** |
| Client (iOS Mobile Device) | 01, 02, 03, 04, 05, 06, 07, 08, 09, 10 |
| Server (Parse Web Service) | 01, 02, 03, 04, 05, 06, 07, 08, 09, 10 |
| Database (Parse Data Service)  \*The Database content is only updated | 01, 02\*, 03\*, 04, 05\*, 06, 07\*, 08, 09,10\* |