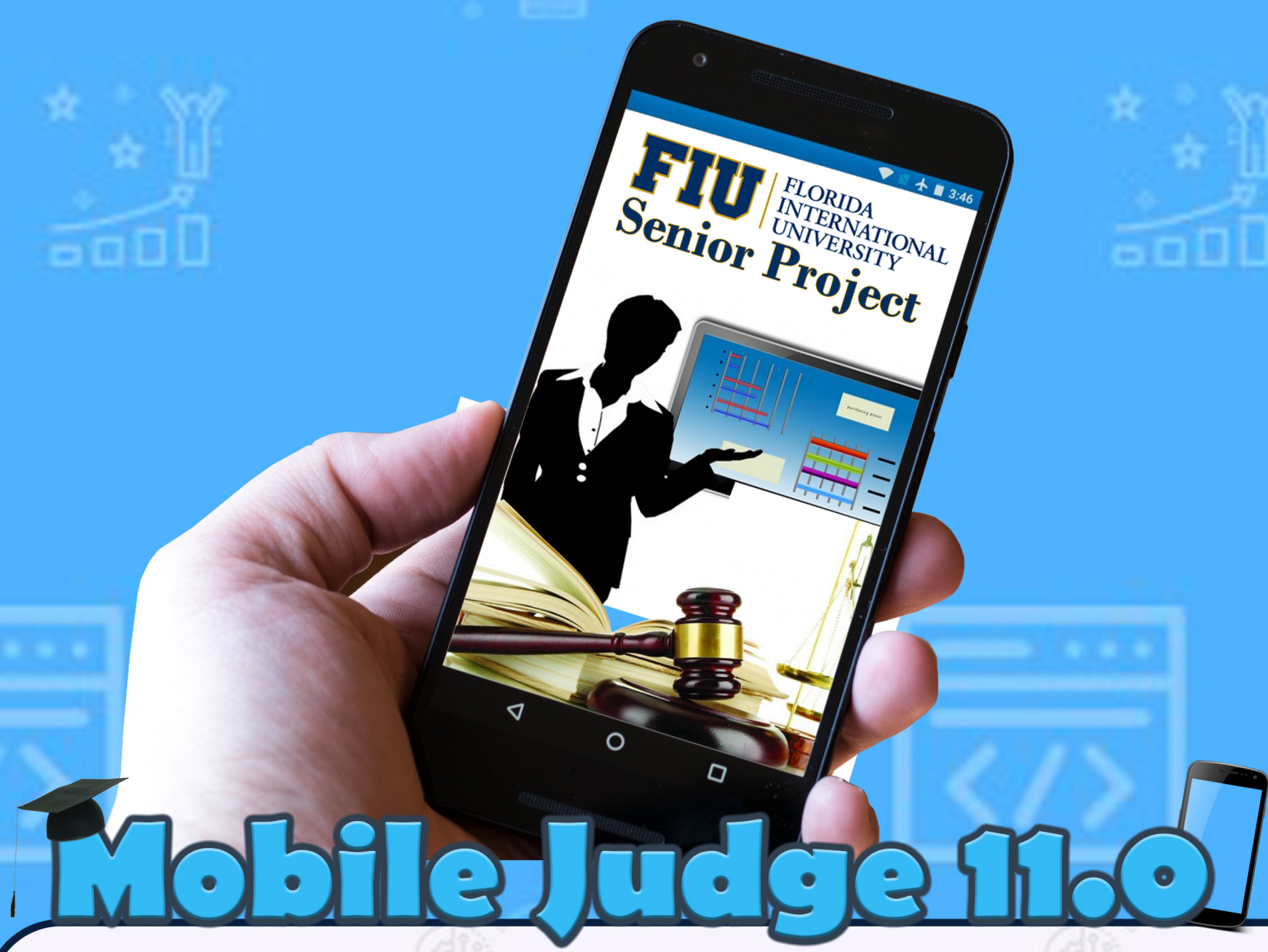


FIU Computing & Information Sciences

Fall 2018 Senior Design Project

Mobile Judge 11.0: Data synchronization
Student: Carlos Larrauri
Menthor: Dr. Masoud Sadjadi, Florida International University
Instructor: Dr. Masoud Sadjadi, Florida International University



Mobile Judge 11.0

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Problem:

Every semester undergraduate students are presented with a project to challenge their achieved knowledge through an academic program. These projects are showcased in front of professors, alumni and other professionals to be judge of their work.

Mobile judge is being presented as a tool to facilitate this process.

The application is completely functional however the way it obtains and store the information can be improved.

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Current System:

The current system is fully functional, however there are some features that can be improved to achieve a better performance.

Due to time constraints and issues capturing the user stories from the client, the current system has yet to be modified to improve these aspects.

The proposal given could be implemented on future versions to achieve a more efficient database and synchronization process.

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Requirements:

The user needs to be able to synchronize the information on the application database with the one provided on the API endpoint.

The user needs to receive a confirmation dialog with the amount of changes to be done on the database.

The user needs to be able to update the information on the database once the confirmation is approved.

The user needs to be able to cancel the changes in case he decides to do it.

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

System Design:

Client Server Model:

Model - view - controller:

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Object Design:

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Implementation:

Sencha ExtJS: used to render the views across multiple platforms.

Sencha CMD: used to compile the changes to the application.

MySQL/phpMyAdmin: DBMS used for database manipulation.

MPN: Used to manage the required node modules.

NGINX: Server used for NodeJS and Oauth.

Oauth: Used for third party authorization.

Vagrant/Ubuntu: Development environment.

Sublime Text: text editor used to edit the code.

POSTMAN: Used to test rest connections to APIs

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Verification:

Test case ID: 744-1

Description/Summary of the Test: Synchronize data between the application database and the API endpoint

Pre-condition: User must be logged in on the application, API endpoint needs to be functional and accessible and the Sync button needs to be pressed.

Expected Result: The data will be synchronized, and the user will be asked to confirm the import.

Actual Result: Due to differences in the database architecture, the incoming data is not synchronized, it can only be imported and redirected to the corresponding tables.

Status: Fail

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Screenshots:

Mobile Judge

- Problem
- Current System
- Requirements
- System Design
- Object Design
- Implementation
- Verification
- Screenshots
- Summary

Summary:

Mobile Judge has been a great tool to manage the senior project showcase, version 11.0 focused on the improvement of the synchronization between the APIs information and the application's database. Some improvements to the database were done and some new improvements are still required to optimize this process.

The development cycle of this functionality presented a series of challenges, associated mostly with the use of new and unknown technologies but overall it provided an opportunity to perceive how working in the industry is.

Acknowledgement

The material presented in this poster is based upon the work supported by Dr. Masoud Sadjadi. I thank Manuel de Armas and Pedro Ramos for assistance, cooperation and mentorship that I received throughout this process.