# Kyle Raja

CS 255

10/01/2022

# CS 255 Business Requirements Document Template

## System Components and Design

### Purpose

* The client is DriverPass
* DriverPass would like an application that helps users schedule appointments and take exams for their drivers license. This should help reduce staff needed at the DriverPass location and make the process more easy for users since most of their testing can be done online until their driving test.

### System Background

* They would like to make the testing process for drivers license more accessible for uses by creating a website application that allows students to take exams online.
* They hope that this will improve the students overall pass rate and experience of the process.
* They would like to fix problems with accessibility for the students by making the experience done more online. They would also like to handle all appointment setting on the web app.
* The components needed will first and foremost be security for the user to login. Next would be exam taking and appointment setting. Each of those could have complex features and more work can be added to the application as time goes one.

### Objectives and Goals

* The application should be 100% accessible online through its website.
* It allow uses to schedule their appointments, make any changes to their appointments, and have assurance that their spot is secured.
* The website should also allow students to take online exams whenever they are required to.
* In order to accomplish these tasks, the team must be organized and create proper planning steps for each department and how everyone will work together to achieve an overall goal.
* I recommend setting up diagrams, specifically process and object diagrams to help the various teams.
* A Gantt chart should also be created so everyone is on the same page on how the departments are going to work together and the timeline of the smaller goals.

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

* The environment will be a web application/website.
* Performance must be fast which means there should be no lagging, crashes, etc.
* Frequent releases need to be applied to the website to fix any bugs or add any functionalities.

#### Platform Constraints

* The web app should be accessible across multiple platforms and not be constrained to just one platform. I recommend that it be accessible on windows, mac, and linux.
* As far as the storage goes, our servers must be able to handle the disk space needed, however we should look into cloud storage as this would improve overall performance as well.

#### Accuracy and Precision

* Users will be given the chance to create a username and password that will be required on the login screen. As the application becomes more popular, I recommend requiring more complex usernames and passwords so that we can insure security.
* Basically if there are breaches towards a users information, the admi should be informed. Also when unusual activity is occurring on the application like an overload of traffic, the admin should analyze the servers to see what is going on.

#### Adaptability

* Framework code should help users be able to change things like username and password without the development team having to change anything in the code.
* Updates to the platform will require skilled hands to update the configurations for the application. For example, if there are framework upgrades, this would require a lot of configuration changes.
* The IT admin needs to have access to all of the configuration and deployments for the applications. They need to also have access to the server base and databases. This helps them see how things are done in the pipeline and ensure security.

#### Security

* The requirements for logging in are a standard username and password.
* The user information can be cross checked in the database through the code to verify that the information is valid.
* If we are using cloud storage, that I believe would handle the client/server aspects of it.
* If the account is being hacked, it should be locked. Typically if the account is trying to be accessed multiple times without success, it should be locked for a given time and the admin team must be notified.
* If the user forgets their password, they should have an option to reset it and have a security code emailed to them to be used on the reset password screen for the application.

### Functional Requirements

* The system shall have a login screen and allow user to input their credentials
* The system shall be able to allow users to schedule and modify their appointments.
* The system shall be ale to allow users to take exams.
* The system should have good performance.

### User Interface

* The needs for the interface are areas where appointments can be set, exams can be take, and any modifications to those areas can be done.
* Also, areas where user profiles can be updated.
* The user should be able to book their appointments, change their appointments, and take their online exams.
* The user should be able to access different types of plans.
* Using the site on pc or mobile should relatively be the same experience.

### Assumptions

* We are assuming that we have all of the technology and staff needed to create this web application.
* We are assuming that we have enough funds to produce the expected results in the timeframe.
* We are assuming that it will work on both pc and mobile platforms.

### Limitations

* I think we have limited staff that could make the web app work on mobile.
* Difficulties in implementing everything to cloud storage.
* Making the process as effective as it was before the webapp was created.
* Security and validating exams.

### Gantt Chart

This is the scale:

Yellow) Planning and Designing

Orange) Development

Red) Business (Product manager, Scrum team)

Blue) QA and Testing

Green) Release and Review

Chart

Description automatically generated with low confidence