**Software Requirements and Design Document**

**For**

**Business Survey Analytics**

Version 1.3

**Authors**:

Samuel Anderson

Michael Fogarty

Joseph Fields

Trey Hamilton

Luis Corps

# Overview (5 points)

*Give a general overview of the system in 1-2 paragraphs (similar to the one in the project proposal).*

We want to provide a business, or any user, with a platform to generate their own survey. For example, the business can create their own personalized survey with specific questions tailored to their business. The platform would then store these results in a database and be able to supply the business with analytics so they can decide what changes should be made to better serve their clients.

# Functional Requirements (10 points)

*List the* ***functional requirements*** *in sentences identified by numbers and for each requirement state if it is of high, medium, or low priority. Each functional requirement is something that the system shall do. Include all the details required such that there can be no misinterpretations of the requirements when read. Be very specific about what the system needs to do (not how, just what). You may provide a brief design rationale for any requirement which you feel requires explanation for how and/or why the requirement was derived.*

***Red text indicates that a functional requirement was modified***

\*\*\* Inc1 functional requirements \*\*\*

* When entering the app both customers and survey takers will be prompted to login or take a survey, this is our home page (High Priority). *This is the first thing that customers see when entering the web app it is important that the home page is clear and concise.*
* Each page will have a link back to the home page (Medium Priority). *It is Important that customers can easily navigate through the web app without having to relaunch the app to get back to the home page.*
* If users choose to take a survey they will be asked to enter a survey code (Medium Priority). *For this increment our main priority is to have the login menu working properly.*
* When the code is submitted, and if it’s correct, it’ll take them to the survey to complete. (High Priority) *This builds off the last requirement Modified in inc3*
* An invalid code will reload the page with a message displayed “invalid code”. (Low Priority) *This builds off the last requirement would be helpful but not necessary Modified in inc3*
* If users choose to login they will be prompted to enter a username(email) however if they don't have an account they will be asked to sign up. Once email is entered they will be sent a code to confirm their account (High Priority) *This is a crucial aspect of the login page given that new customers will not have an account yet need one in order to start creating surveys. Modified in inc3*
* New users will link them to a registration page where they will be prompted for certain information that will create their account. (High Priority) *This builds off the last requirement and is where the customer's will establish their email, password, and company name.*
* *Users who will chose to create a survey will have the ability to create unique multiple choice question.*(High Priority)
* Once they have finished creating the survey they will be given the option to save and publish after this they will be given their survey code.(High Priority) *this is a core functionality of the web app.*
* If a company/user has pre-existing surveys they can view the results for their surveys and see what the most popular answer choice for each question is. (High Priority) *without this feature companies would have no way of interpreting the results of their surveys which would make the creation of a survey borderline useless*
* No two Accounts can have access to the same survey, all customer accounts are independent of each other.

\*\*\* Inc2 functional requirements \*\*\*

* When a survey taker enters a survey code they will be redirected to a new page where they can begin the survey.
* When a survey taker selects an answer choice it will be stored in the database.(High Priority) *This is how our customers will be able to view the results from their surveys.*
* Survey takers can only view and take one survey at a time.(High Priority) *When a survey taker types in a survey code it should only bring up the questions and answers for that survey.*

*.*

\*\*\* Inc3 functional requirements \*\*\*

* New users will be prompted to create a password, email, first / last name, and company name. *This is a modification of an inc1 functional requirement*
* If a user has an account they can then login with their email and password.*This is a modification of an inc1 functional requirement*
* In case of an incorrect password users will be prompted to re-attempt to login.*This is a modification of an inc1 functional requirement*
* *Customers will be prompted for the type of question they would like to create based on the types of answer choices we chose to make available. This is to decrease chances of confusion and make our service simple and easy to use.*
* *Created surveys are available for taking based on a generated survey ID that survey takers will type in to be directed to the appropriate survey.*
* *Business owners will have access to only their own completed surveys, as well as the results of those surveys (High Priority)*

***\*NOTE***

*Security is a known issue and is something that we would attempt to fix in the*

*next increment. The reason for not using Auth0 were time constraints, we felt that*

*having at least an email and password would create a minimal level of security.*

*Also our primary goal was for users to create and see survey results so we felt*

*that working on those requirements took precedence over the login page.*

# Non-functional Requirements (10 points)

*List the* ***non-functional requirements*** *of the system (any requirement referring to a property of the system, such as security, safety, software quality, performance, reliability, etc.) You may provide a brief rationale for any requirement which you feel requires explanation as to how and/or why the requirement was derived.*

***Red text indicates that a non-functional requirement was modified***

\*\*\* Inc1 non-functional requirements \*\*\*

* Home page will take no longer than 2 seconds to load.
* To move from one page back to another ex. sign up page **-->** home page should also take no longer than 2 seconds to load.
* Database will store a unique token from the user to determine if they have an account. *We plan to use Auth0 to ensure a secure login for customers instead of rolling out our own, which due to our lack of experience would likely be much less secure.*
* Login Information will take no longer than 3 seconds to be sent to the database
* Every user will only have access to their own information.
* Login page is secured using Auth0.

\*\*\* Inc2 non-functional requirements \*\*\*

* It will take to longer than 2 second to retrieve a customers or surveys information from the database
* It will take to longer than 2 second to add a customers or surveys information to the database
* Neither customers or survey takers will have direct access to the database. (High Priority) *This will prevent customers or survey takers from changing their customer \_ID, survey\_ID, or any other important information we may need as well as protecting the privacy of our users.*

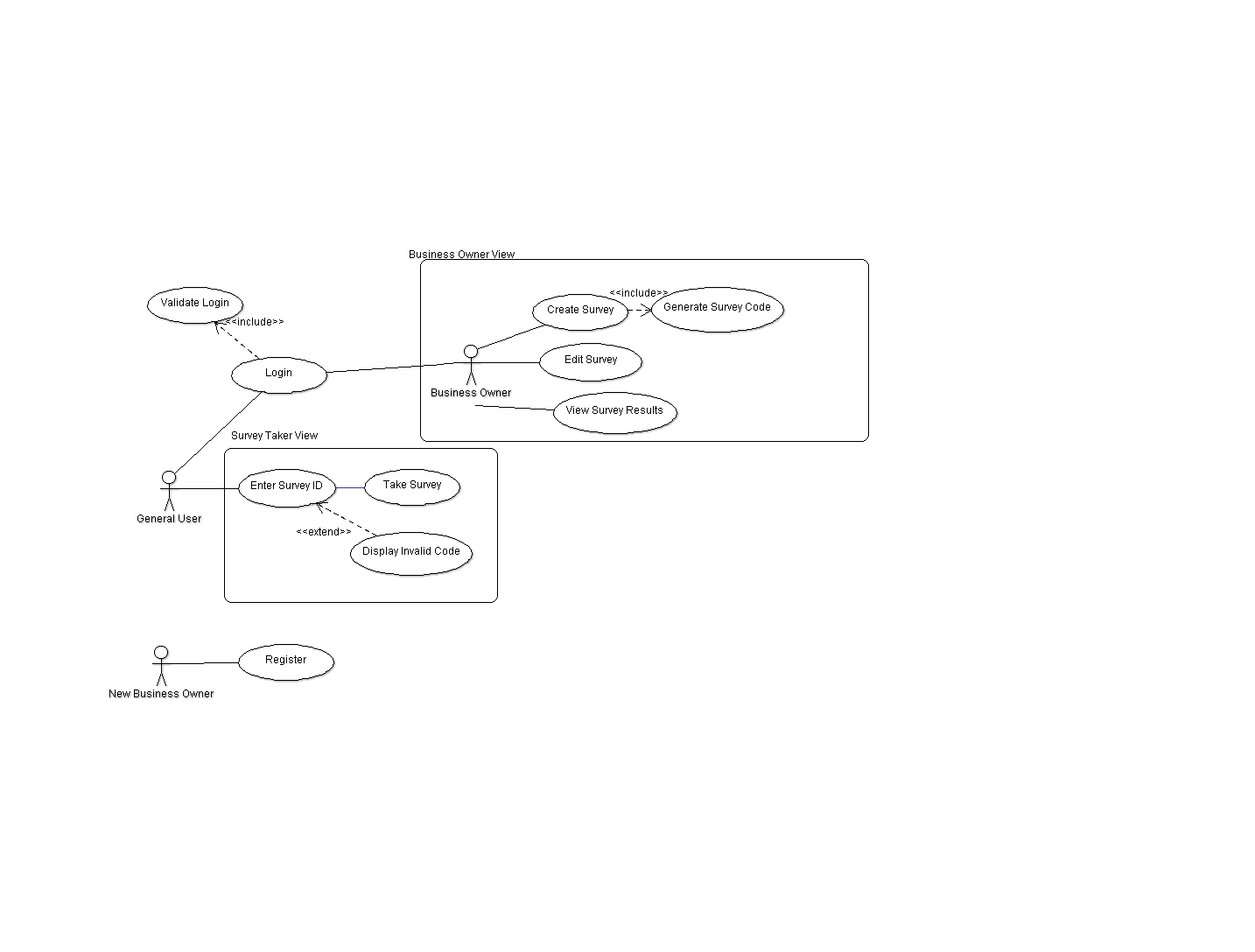
\*\*\* Inc3 non-functional requirements \*\*\*

* Login in page is secured with a username and password. *This is a modification of inc1 non-functional requirement.*
* Each role (admin, business owner, survey taker) is prompted through the site to easily complete their desired task, whether it be business owners creating a survey or viewing results, or survey takers completing surveys.
* Customers and survey takers have no access to modifying database information, only creating accounts and surveys.
* Communication with database is done in a timely manner.

# Use Case Diagram (10 points)

*This section presents the* ***use case diagram*** *and the* ***textual descriptions*** *of the use cases for the system under development. The use case diagram should contain all the use cases and relationships between them needed to describe the functionality to be developed. If you discover new use cases between two increments, update the diagram for your future increments.*

***Textual descriptions of use cases****: For the first increment, the textual descriptions for the use cases are not required. However, the textual descriptions for all use cases discovered for your system are required for the second and third iterations.*

**

# Class Diagram and/or Sequence Diagrams (15 points)

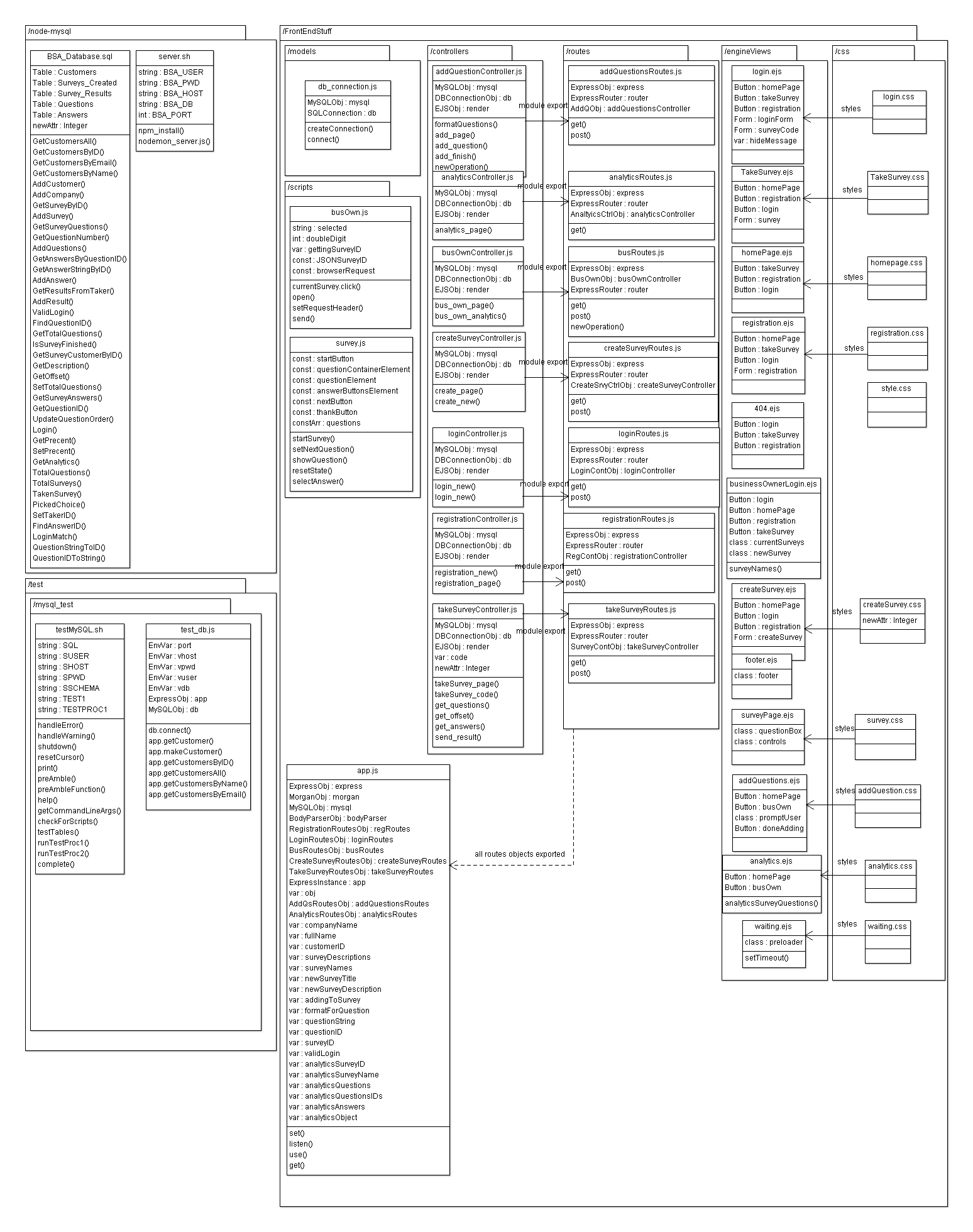
*This section presents a high-level overview of the anticipated system architecture using a* ***class******diagram*** *and/or* ***sequence diagrams****.*

*If the main* ***paradigm*** *used in your project is* ***Object Oriented*** *(i.e., you have classes or something that acts similar to classes in your system), then draw the* ***Class Diagram******of the entire system and Sequence Diagrams for the three (3) most important use cases in your system.***

*If the main* ***paradigm*** *in your system is* ***not Object Oriented*** *(i.e., you* ***do not*** *have classes**or anything similar to classes in your system) then only draw* ***Sequence Diagrams****,* ***but for all the use cases of your system.*** *In this case, we will use a modified version of Sequence Diagrams, where instead of objects, the lifelines will represent the functions in the system involved in the action sequence.*

***Class Diagrams*** *show the* ***fundamental objects/classes*** *that must be modeled with the system to satisfy its requirements and* ***the relationships*** *between them. Each class rectangle on the diagram* ***must also include the attributes and the methods of the class*** *(they can be refined between increments). All the* ***relationships between classes and their multiplicity*** *must be shown on the class diagram.*

*A* ***Sequence Diagram*** *simply depicts* ***interaction******between objects*** *(or* ***functions -*** *in our case - for non-OOP systems) in a sequential order, i.e. the order in which these interactions take place. Sequence diagrams describe how and in what order the objects in a system function.*

**

# Operating Environment (5 points)

BSA is a web-based application, so it is necessary for the user to have access to the server/host where the files are located. Subsequently, the user must have a web browser that supports HTML5, CSS3, and JavaScript. BSA can operate on any operating system that can meet the web browsing requirements.

# Assumptions and Dependencies (5 points)

Dependencies

* Currently hosting database on localhost, which must be functioning properly to send and receive data.
* Third-party modules in use: express, morgan, mysql, body-parser, express-myconnection