

**Computer Science and Engineering**

**NYU Student Club Event Planner**

**Systems Analysis Specification**

**Version 1.0**

Document Number: SAS-001

Project Number: A10

Project Members:

Gryczka, Patrick

Han, Qiao Yang

Huang, Alan

Papa, Kostaq

**REVIEW AND APPROVALS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Printed Name**  **and Title** | **Function (Author, Reviewer, Approval)** | **Date** | **Signature** |
| Professor Strauss | Instructor | April 14, 2014 |  |
| Patrick Gryczka | Author | April 14, 2014 | On file |
| Qiao Yang Han | Author | April 14, 2014 | On file |
| Alan Huang | Author | April 14, 2014 | On file |
| Kostaq Papa | Author | April 14, 2014 | On file |

**REVISION LEVEL**

|  |  |  |
| --- | --- | --- |
| **Date** | **Revision Number** | **Purpose** |
| **April 9, 2014** | **Version 1.0** | **Initial Release** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**TABLE OF CONTENTS**

**1. INTRODUCTION...........................................................................................1**

**1.1 PURPOSE..........................................................................................1**

**2. SCOPE..........................................................................................................1**

**2.1 IDENTIFICATION……………………………………………………………………………..1**

**2.2 BOUNDS...........................................................................................1**

**2.3 OBJECTIVES......................................................................................1**

**2.4 SYSTEM OVERVIEW..........................................................................2**

**2.5 DOCUMENT OVERVIEW....................................................................2**

**3. REFERENCE DOCUMENTS.............................................................................2**

**4. BUISNESS REQUIREMENTS...........................................................................3**

**4.1 TECHNOLOGY…………………………………………………................................3**

**4.2 ECONOMICS……………………………………………………...............................3**

**4.3 REGULATORY AND LEGAL………………………………………………................3**

**4.4 MARKET CONSIDERATIONS.............................................................3**

**4.5 RISKS AND ALTERNATIVES...............................................................3**

**4.6 HUMAN RESOURCES AND TRAINING...............................................4**

**5. LOGICAL ARCHITECHTURAL SPECIFICATION.................................................4**

**5.1 CONTEXT DIAGRAM…………………………………………..............................4**

**5.2 SYSTEM CAPABILITY REQUIREMENTS………………………………...............4**

**5.2.1 CAPABILITIES……………………………………………................................4**

**5.2.2 USE CASE DIAGRAMS……………………………....................................6**

**5.2.3 USE CASE DESCRIPTIONS………………………………….........................8**

**5.3 USER INTERFACE REQUIREMENTS……………………………………………….....11**

**5.4 COMPONENT (COMPONENT/PACKAGE/SUBSYSTEM)**

**ARCHITECTURE……………………………………………………………….....…………….....11**

**5.4.1 COMPONENT DESCRIPTIONS…………………………………………………....11**

**5.4.2 COMPONENT ARCHITECTURE DIAGRAM…………………………………..12**

**5.5 CLASS DIAGRAMS……………………….....................................................13**

**5.6 CLASS RELATIONSHIP/INTERACTION DIAGRAMS…………....................13**

**5.7 EVENTS……………………………………….....................................................14**

**5.7.1 MOTIVES………………………………………............................................14**

**5.7.2 EVENT DIAGRAMS…………………………...........................................15**

**5.8 ACTIVITY/STATE (SCENARIO) SECTION (TO BE COMPLETED IN DESIGN)**

**.............................................................................................................16**

**5.9 STATE LOGIC (TO BE COMPLETED IN DESIGN)………….......................16**

**5.10 BEHAVIOR…………………………………………..........................................16**

**5.10.1 SEQUENCE DIAGRAMS……………………......................................16**

**5.10.2 COLLABORATION DIAGRAMS……............................................17**

**5.11 DICTIONARIES (INITIATED HERE AND COMPLETED IN DESIGN)......18**

**6. NON-FUNCTIONAL/OPERATIONAL SPECIFICATIONS……..............................18**

**6.1 SYSTEM EXTERNAL INTERFACE REQUIREMENTS..............................18**

**6.2 SAFETY REQUIREMENTS..................................................................18**

**6 .3 SECURITY AND PRIVACY REQUIREMENTS......................................18**

**6.4 SYSTEM ENVIRONMENT REQUIREMENTS…….................................18**

**6.5 COMPUTER RESOURCE REQUIREMENTS…………..............................18**

**6.5.1 COMPUTER HARDWARE REQUIREMENTS……...........................18**

**6.5.2 COMPUTER HARDWARE RESOURCES REQUIREMENTS.............19**

**6.5.3 COMPUTER SOFTWARE REQUIREMENTS……….........................19**

**6.5.4 COMPUTER COMMUNICATIONS REQUIREMENTS….................19**

**6.6 SYSTEM QUALITY FACTORS………....................................................19**

**6.7 DESIGN AND CONSTRUCTION CONSTRAINTS……………....................19**

**6.8 PRESONNEL-RELATED REQUIREMENTS…………................................19**

**6.9 TRAINING-RELATED REQUIREMENTS……………................................19**

**6.10 LOGISTICS-RELATED REQUIREMENTS………….................................20**

**6.11 PACKAGING REQUIREMENTS………….............................................20**

**6.12 PRECEDENCE AND CRITICALITY REQUIREMENTS………...................20**

**6.13 OTHER REQUIREMENTS………………..............................................20**

**7. SYSTEM TEST PLAN REQUIREMENTS.........................................................20**

**8. QUALIFICATION PROVISIONS....................................................................20**

**9. REQUIREMENTS TRACEABLITIY.................................................................21**

**10. RATIONALE.............................................................................................21**

**11. NOTES....................................................................................................21**

**12. APPENDICES...........................................................................................21**

**12.1 DICTIONARIES.............................................................................21**

**12.2 UML DIAGRAMS IF NOT INCLUDED IN THE BODY OF THE DOCUMENT**

**..........................................................................................................23**

**12.3 SCHEDULE TRACKING………………………….......................................24**

**12.4 DEFECT TRACING…………………......................................................24**

**1. INTRODUCTION**

**1.1 PURPOSE**

The purpose of this document is to specify the business and user requirements for the event planning system, which include the function and non-functional requirements, and the system requirements. The requirements are analyzed, and test cases are derived for the system. The intended audience of this document is the client, the business analysts, the quality assurance team and the developers.

**2. SCOPE**

**2.1 IDENTIFICATION**

**NYU Student Club Event Planner, Team A10, SAS, Version 1.0, 4/23/2014**

**2.2 BOUNDS**

The bounds of the system are an application server, web-based user interface to the system and a database to store the system data.

**2.3 OBJECTIVES**

This project will follow an incremental life cycle.

Initial deliverables will be as follows

* + Software Requirement Specification (SRS) - 4/8/2014
  + Software Project Management Plan (SPMP) - 4/10/2014
  + Software Analysis Specification (SAS) - 4/14/2014
  + Software Design Document (SDD) - 4/28/2014

**2.4 SYSTEM OVERVIEW**

The purpose of the NYU Student Club Event Planner is to create a better means of communication between the clubs at NYU and the NYU student body. This software proposes to fill this perceived deficit with a social infrastructure that should be able to both connect students with each other and keep them informed on events.

All activities directly related to the purpose are considered to be within the scope. All activities not directly related to the purposes are considered to be out of scope. For example, issues concerning the finance and administration of student club are not within the scope of this project.

**2.5 DOCUMENT OVERVIEW**

**The System Analysis Specification outlines key specifications and requirements of the NYU Student Club Event Planner. Section 1, Introduction, outlines the purpose of the SAS. Section 2, Scope, specifies the bounds and objectives of the system. Section 3, Reference Documents, specifies referenced documents. Section 4, Business Requirements, specifies the financial needs and considerations. Section 5, Logical Architecture Specification, outlines the system; it includes context, use cases, user interface needs, and class diagrams and relationships. Section 6, Non-functional/Operational Specifications, details the systems requirements, ranging from hardware to personnel training. The rest of the sections are deemed sufficiently explained by their titles.**

**3. REFERENCE DOCUMENTS**

NYU Student Clubs Event Planner, Team A10, Project Proposal, Version 1.0, 4/1/2014

NYU Student Clubs Event Planner, Team A10, System Requirement Specifications, Version 1.1, 4/8/2014

NYU Student Clubs Event Planner, Team A10, System Project Management Plan, Version 1.0, 4/10/2014

**4. BUISNESS REQUIREMENTS**

**4.1 TECHNOLOGY**

There is a need for an infrastructure that would better allow the student clubs of NYU to connect with the students, and keep them informed about the club’s activities. Beyond the user interface the system will need to be capable of retrieving information from outside databases and maintaining its own databases accordingly in order to be useful.

**4.2 ECONOMICS**

In order for the system to be profitable, it will need advertisers. Whether this advertising will take the form of ad placement within the user interface or a sponsor name incorporated into the branding of the system will depend on whether or not we will be able to procure a dedicated sponsor. In addition, for cost efficiency, the system will need to be reliably automated to avoid maintenance or management costs.

**4.3 REGULATORY AND LEGAL**

**None, as of the writing of this document.**

**4.4 MARKET CONSIDERATIONS**

While our users will be NYU students, our customers will be the club organizations looking to advertise their events and organizations to our user base.

**4.5 RISKS AND ALTERNATIVES**

The greatest risk we face is that clubs may not see the benefit of using our system. Accordingly we will need to ensure that the project is appropriately represented as an opportunity for them to better connect with potential club members and existing ones. This risk is expected to decrease as the number of clubs taking advantage of our system increases.

**4.6 HUMAN RESOURCES AND TRAINING**

The development team needs training in Graphical User Interface development. While the team has a solid grasp of the programing aspects of the project, they will need to better acquaint themselves will the creation of user interfaces in order to make a system that users will find useful and easily understandable. The developers will also need training for the following technologies: Java, HTML, CSS, JavaScript, C++, and SQL. They must learn how to use C++ to communicate with the database. The developers may also get training for Microsoft SQL Server and Visual Studio.

**5. LOGICAL ARCHITECHTURAL SPECIFICATION**

**5.1 CONTEXT DIAGRAM**

users

**5.2 SYSTEM CAPABILITY REQUIREMENTS**

**5.2.1 CAPABILITIES**

1. Potential clients must register on the website as club leaders or non-club members.

1.1 The two types of clients must use their NYU email to register and the

system will confirm they are NYU students.

1.2 Email is sent to the clients for confirmation.

1.3 The clients are directed to create a username and password.

2. The clients must be logged in.

2.1 Username should be at least 5 characters excluding special characters

(Only ASCII characters).

2.2 Password should be at least 8 characters including at least one number, one

Letter and one special character.

2.3 The club leaders are taken to a portal with different privileges from those of

the non club members. The club leaders can accept requests from the non

club members to be part of their club.

3. Non club members and club member can search and select to attend events. Club

leaders can create events.

3.1 If non club members want to attend an event, they will click on it and will be

taken to a page that contains more information about the event.

4. Club leaders can edit an event.

4.1 If the club leader wants to create an event then he clicks in create event tab

in the main page and will be taken over to a page where they can add, edit

or delete events.

4.2 Once an event was created only the club leader can still edit it.

5. Attendance notification is generated.

5.1 Once the club members or non club members accept to attend an event, the

club leader will get a notification.

6. Possible event locations are given.

6.1 A map of the NYU campuses are provided so the clients attending the

event can easily find the location.

**5.2.2 USE CASE DIAGRAMS**

****

**5.2.3 USE CASE DESCRIPTIONS**

**Individual Looking for Club Information**

|  |  |  |
| --- | --- | --- |
| **View today’s events** | | |
| **Description** | User – Individual Looking for Club Information, can view posted club events | |
| **Pre-Condition** | User is not logged in user interface. | |
| **Flows** | **Basic or Normal Flows** | 1. User browse through all the posted club events.  2. User can search certain events by specifying the date, the time and the location of the event.  3. User can click into a selected event and read the details of a specific event.  4. User can look into the club which is organizing the event. |
|  | **Alternative Flows** | N/A |
| **Special**  **Requirements** | N/A | |
| **Extension**  **Points** | N/A | |

|  |  |  |
| --- | --- | --- |
| **View all clubs** | | |
| **Description** | User – Individual Looking for Club Information, can view all clubs in the website. | |
| **Pre-Condition** | User must have logged in user interface. | |
| **Flows** | **Basic or Normal Flows** | 1. User browse through all the clubs.  2. User can search certain club by specifying the club’s name.  3. User can click into a selected club and read the information of the club. |
|  | **Alternative Flows** | N/A |
| **Special**  **Requirements** | N/A | |
| **Extension**  **Points** | N/A | |

|  |  |  |
| --- | --- | --- |
| **View all upcoming events** | | |
| **Description** | User – Individual Looking for Club Information, can view all upcoming events in the website. | |
| **Pre-Condition** | User must have logged in user interface. | |
| **Flows** | **Basic or Normal Flows** | 1. User browse through all upcoming events.  2. User can search certain event by specifying the name, the date, the time or the location of the event.  3. User can click into a selected event and read the description of the event. |
|  | **Alternative Flows** | N/A |
| **Special**  **Requirements** | N/A | |
| **Extension**  **Points** | N/A | |

|  |  |  |
| --- | --- | --- |
| **View all the tracked events** | | |
| **Description** | User – Individual Looking for Club Information, can view all the tracked events. | |
| **Pre-Condition** | User must have logged in user interface. | |
| **Flows** | **Basic or Normal Flows** | 1. User browse through all the tracked events.  2. User can search certain tracked event by specifying the name, the date, the time or the location of the event.  3. User can click into a selected tracked event and read the description of the tracked event. |
|  | **Alternative Flows** | N/A |
| **Special**  **Requirements** | N/A | |
| **Extension**  **Points** | N/A | |

**Club Official Managing Activities and Events for a club**

|  |  |  |
| --- | --- | --- |
| **Schedule new events** | | |
| **Description** | User - Club Official Managing Activities and Events for a club, can schedule new club events to encourage students to join. | |
| **Pre-Condition** | User must have logged in user interface for managing users. | |
| **Flows** | **Basic or Normal Flows** | 1. User selects to schedule a new club event.  2. User specifies the date, the time and the location of the event.  3. User writes down the name of the event and the name of the host.  4. User describes the details of the event.  5. User uploads an advertising picture of the event. |
|  | **Alternative Flows** | N/A |
| **Special**  **Requirements** | N/A | |
| **Extension**  **Points** | N/A | |

|  |  |  |
| --- | --- | --- |
| **Modify Club’s Description** | | |
| **Description** | User - Club Official Managing Activities and Events for a club, can modify the description of their club | |
| **Pre-Condition** | User must have logged in user interface for managing users. | |
| **Flows** | **Basic or Normal Flows** | 1. User selects to modify the club’s description  2. User can rename their club’s name, change the club’s member size.  3. User can modify the club’s theme.  4. User can change the name of the club officials. |
|  | **Alternative Flows** | N/A |
| **Special**  **Requirements** | N/A | |
| **Extension**  **Points** | N/A | |

|  |  |  |
| --- | --- | --- |
| **Update Existing Events** | | |
| **Description** | User - Club Official Managing Activities and Events for a club, can update the existing club events | |
| **Pre-Condition** | User must have logged in as a Managing User. | |
| **Flows** | **Basic or Normal Flows** | 1. User selects to update the existing club’s events.  2. User selects the specific event he/she wants to update from a list of existing events.  3. User can modify the date, the time and the location of the event.  4. User can rename the event and the host.  5. User can modify the details of the event.  6. User can upload a new advertising picture of the event. |
|  | **Alternative Flows** | N/A |
| **Special**  **Requirements** | N/A | |
| **Extension**  **Points** | N/A | |

**5.3 USER INTERFACE REQUIREMENTS**

**The user interface will need to allow all users to log in and log out. Club administrators to access their club’s information, which is comprised of a name, description, and a list of events. Administrators will also be able to edit each of those data members, changing names, and adding and removing events. Users who are club members must be able to view clubs, view events, track all the events of a given club, and track individual events.**

**5.4 COMPONENT (COMPONENT/PACKAGE/SUBSYSTEM)**

**ARCHITECTURE**

**5.4.1 COMPONENT DESCRIPTIONS**

**1. User Interface**

The user interface is given to the user to interact with the system. This includes the aesthetic appearance of the website, response time, and the content that is presented to the user.

**2.** **Web Server**

The web server is the software that delivers web content which is accessed through the internet. Its function is output the HTML documents that the user is trying to access. This is done using the HTTP protocol. The user will request a document in the form of a URL and the server will respond with the document.

**3.** **Application Server**

The application server handles all application operations between the users and the backend database. This includes socket programming between the client and server and the middleware API for accessing the database.

**5.4.2 COMPONENT ARCHITECTURE DIAGRAM**

**User Interface**

**Database Server:** *Microsoft SQL Server*

**Web Server**

Event Planner System Database

PHP web pages using

HTML, CSS, JavaScript

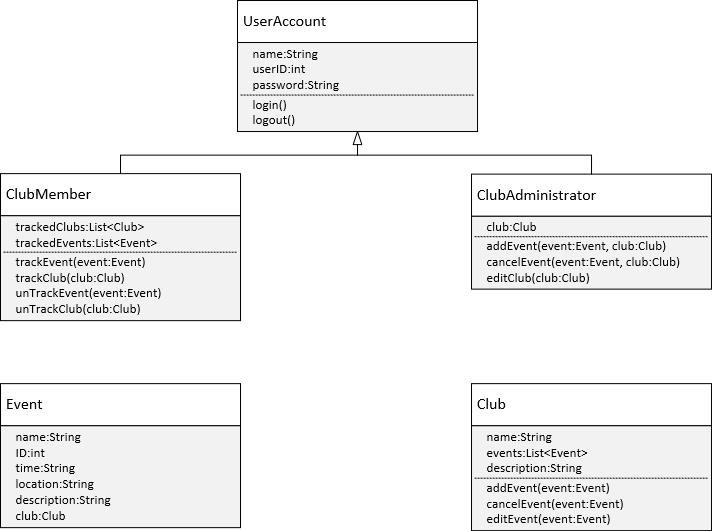
**Application Server:** *C++, ODBC*

Verify user login information

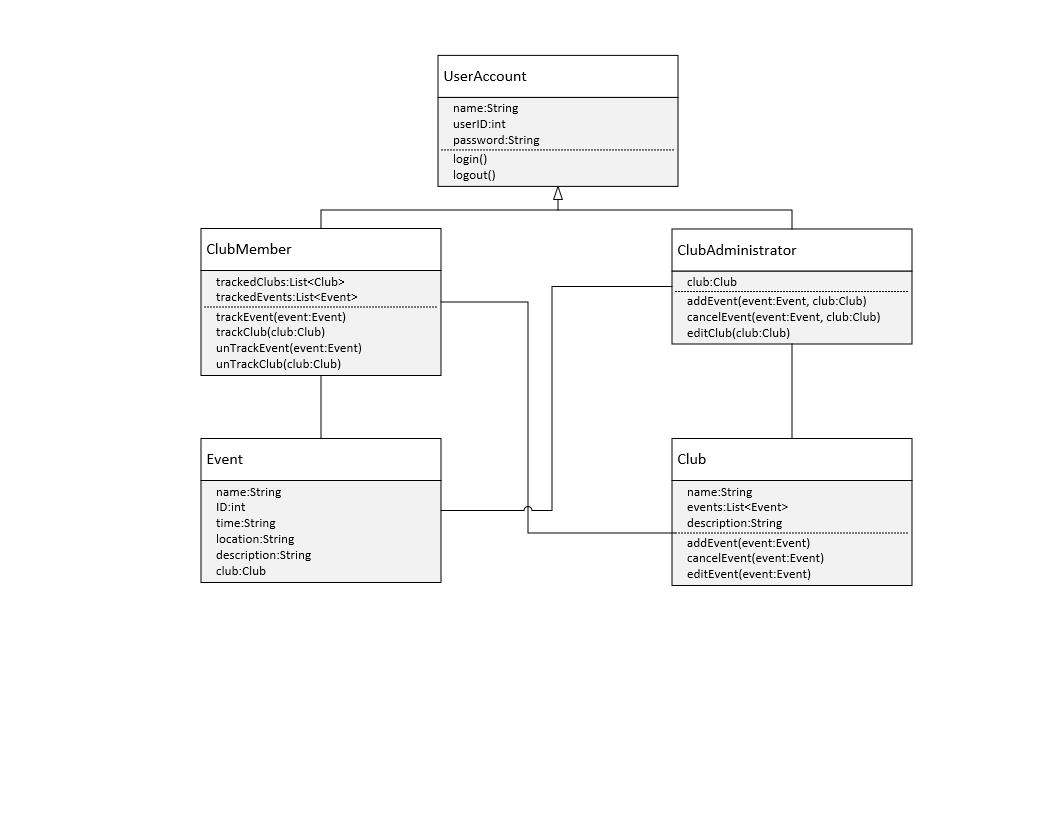
Add/Modify/Delete items

Search for items and view item information

**5.5 CLASS DIAGRAMS**



**5.6 CLASS RELATIONSHIP/INTERACTION DIAGRAMS**

**5.7 EVENTS**

**5.7.1 MOTIVES**

1. User registers in the system

* The user clicks on the register link in the log in screen if they do not have an account.
* The system directs them to the registering page where they will enter their name, email address, desired username and password.
* The system will respond to the user by verifying their email address.
* If the user is not an NYU student, they will not be able to join the system and an error will be generated.

2. User logs into the system

* User will log into the system using their username and password.
* The system will verify their username and password combination.
* If the password and username combination do not match, the user is prevented from logging in and an error message is generated.
* If the login credentials are correct, the user is taken to their home page.

3. User navigating in their home page

* The user will be in their home page and will see the search tab, and a left panel will display events that will happen.
* When the user searches for a certain event or client they will be directed to the results of the search page
* When the user clicks on a category on the left side of the panel, the user is taken to a page displaying all the scheduled events and their information in that category.

4. User navigating in the search results page

* When the user clicks to search a information, they will be taken to the search results page.
* The user can mainly search for an item that is either an event or a profile.
* When the user types in the name of item, the information about the event or profile is displayed.

**5.7.2 EVENT DIAGRAMS**

|  |  |
| --- | --- |
| ****Event**** | ****Response**** |
| ****Login**** | Check if the users login successfully. If it is successfully, then allow the users access to the actions associated with their accounts. Else, return back to the login screen. |
| ****User selects to view today's events**** | The users can browse through all the posted club events. After searching certain events by specifying the date, the time and the location of the event, the users can click into a selected event and read the details of a specific event. The users can look into the club which is organizing the event. |
| ****User selects to view all clubs**** | The users can browse through all the clubs can search certain club by specifying the club’s name. The users can click into a selected club and read the information of the club. |
| ****User selects to view all upcoming events**** | The users can browse through all upcoming events. After searching certain event by specifying the name, the date, the time or the location of the event, the users can click into a selected event an read the description of the event. |
| User selects to view all the tracked events | The users can browse through all the tracked events. After searching certain tracked event by specifying the name, the date, the time or the location of the event, they can click into a selected tracked event and read the description of the tracked event. |
| User selects to schedule new events | The users can select to schedule a new club event. After specifying the date, the time and the location of the event, the users write down the name of the event and the name of the host. Then, they describe the details of the event. They have the option to upload an advertising picture of the event. |
| User selects to modify club's description | The users can select to modify the club's description. User can rename their club’s name, change the club’s member size. The users can modify the club’s theme. The users can change the name of the club official. |
| User selects to update the existing events | The users select to update the existing club's events. The users select to update the existing club’s events. The users selects the specific event he/she wants to update from a list of existing events. The users can modify the date, the time and the location of the event. The users can rename the event and the host. The users can modify the details of the event. The users can upload a new advertising picture of the event |

**5.8 ACTIVITY/STATE (SCENARIO) SECTION (TO BE COMPLETED IN DESIGN)**

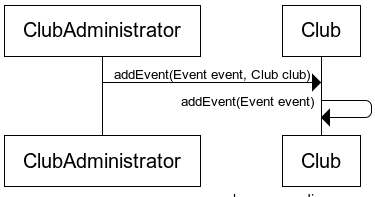
**To be completed in design.**

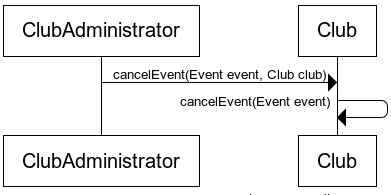
**5.9 STATE LOGIC (TO BE COMPLETED IN DESIGN)**

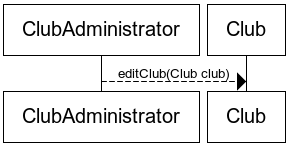
**To be completed in design.**

**5.10 BEHAVIOR**

**5.10.1 SEQUENCE DIAGRAMS**







**5.10.2 COLLABORATION DIAGRAMS**

**To be completed in the design.**

**5.11 DICTIONARIES (INITIATED HERE AND COMPLETED IN DESIGN)**

**See Section 12.1.**

**6. NON-FUNCTIONAL/OPERATIONAL SPECIFICATIONS**

**6.1 SYSTEM EXTERNAL INTERFACE REQUIREMENTS**

**None.**

**6.2 SAFETY REQUIREMENTS**

**Safety precautions will be taken when operating electronic devices and personnel will be trained.**

**6 .3 SECURITY AND PRIVACY REQUIREMENTS**

**User credentials will be stored securely and passwords will be hashed. SQL database will be access securely using prepared statements to prevent SQL injection. On the web interface, XSS will be prevented by encoding and CSRF will be prevented with one time tokens.**

**6.4 SYSTEM ENVIRONMENT REQUIREMENTS**

**Normal conditions will be sufficient.**

**6.5 COMPUTER RESOURCE REQUIREMENTS**

**6.5.1 COMPUTER HARDWARE REQUIREMENTS**

Users will need to have computers for the user interface program and we will need mainframe computers to handle the databases.

**6.5.2 COMPUTER HARDWARE RESOURCES REQUIREMENTS**

Specific hardware resource requirements will not be available until the development process, but they are not expected to be intensive.

**6.5.3 COMPUTER SOFTWARE REQUIREMENTS**

Computer software requirements may evolve as the software is being developed, but currently it is expected that the servers will need to be able to handle fill relational databases, and the users computers will need to be able to run compiled C++ code.

**6.5.4 COMPUTER COMMUNICATIONS REQUIREMENTS**

Users will need to have an internet connection. However, it does not need to be a high speed internet connection.

**6.6 SYSTEM QUALITY FACTORS**

Database manipulations will need to be done correctly to ensure that users do not have problems with event scheduling or joining. User interface will need to be easily understood and aesthetically pleasing.

**6.7 DESIGN AND CONSTRUCTION CONSTRAINTS**

**Determined during design.**

**6.8 PRESONNEL-RELATED REQUIREMENTS**

Personnel should be skilled programmers with experience with database maintenance.

**6.9 TRAINING-RELATED REQUIREMENTS**

The NYU Student Event Planner should not require any training. The user interface should be intuitive enough to be easily understood at first glance.

**6.10 LOGISTICS-RELATED REQUIREMENTS**

**None as of the writing of this document.**

**6.11 PACKAGING REQUIREMENTS**

**None as of the writing of this document.**

**6.12 PRECEDENCE AND CRITICALITY REQUIREMENTS**

**None as of the writing of this document.**

**6.13 OTHER REQUIREMENTS**

**None as of the writing of this document.**

**7. SYSTEM TEST PLAN REQUIREMENTS**

The user interface will be presented to a wide test demographic to see how users respond both to the aesthetics and performance of the system and to deduce what changes would benefit the user interfaces design. The back bone of the system will be tested by Quality Assurance using model internal and external databases to ensure the quality of the database manipulation logic, and to check that the scheduler responds appropriately.

**8. QUALIFICATION PROVISIONS**

This document was written as per an outline provided by our instructor. The instructor will perform an initial review for quality, from which internal quality assessment methods will be derived.

**9. REQUIREMENTS TRACEABLITIY**

**As of the writing of the System Analysis Specification for the NYU Student Event Planner, all the requirements are traceable to the Project Proposal and the System Requirements Specification.**

**10. RATIONALE**

The reasoning behind this system, is that the social dynamics of NYU can be improved by improving the communication between clubs and their current and potential members. We believe that by providing a centralized platform of access to club activities for students, we can make it easier for students to keep up with the club events and meetings and hopefully increase participation

**11. NOTES**

**12. APPENDICES**

**12.1 DICTIONARIES**

|  |  |  |
| --- | --- | --- |
| Class | UserAccount | |
| Parent | None | |
|  | Name | Description |
| Methods | login | Logs the user into the system |
| logout | Logs the user out of the system |
| Attributes | name | The name of the user |
| userID | The identification number of the user |
| password | The user’s password |

|  |  |  |
| --- | --- | --- |
| Class | ClubMember | |
| Parent | UserAccount | |
|  | Name | Description |
| Methods | trackClub | Adds a passed in club to the trackedClubs data member |
| trackEvent | Adds a passed in event to the trackedEvents data member |
| untrackClub | Removes a passed in club from the trackedClubs data member |
| untrackEvent | Removes a passed in club from the trackedEvents data membe |
| Attributes | trackedClubs | List of clubs |
| trackedEvents | List of events |

|  |  |  |
| --- | --- | --- |
| Class | ClubAdministrator | |
| Parent | UserAccount | |
|  | Name | Description |
| Methods | addEvent | Adds the passed in event to the passed in club’s list of events |
| cancelEvent | Removes the passed in event from the passed in club’s list of events |
| editClub | Edits the passed in club |
| Attributes | club | The club that the ClubAdministrator is administrating |

|  |  |  |
| --- | --- | --- |
| Class | Event | |
| Parent | None | |
|  | Name | Description |
| Attributes | name | The name of the event |
| ID | The identification number of the event |
| time | The time of the event |
| location | The location of the event |
| description | The description of the event |
| club | The club that the event is created under |

|  |  |  |
| --- | --- | --- |
| Class | Club | |
| Parent | None | |
|  | Name | Description |
| Methods | addEvent | Adds the passed in event to the club’s list of events |
| cancelEvent | Removes the passed in event from the club’s list of events |
| editEvent | Edits the passed in event |
| Attributes | name | The name of the club |
| events | The club’s list of events |
| description | The club’s description |

**12.2 UML DIAGRAMS IF NOT INCLUDED IN THE BODY OF THE DOCUMENT**

**12.3 SCHEDULE TRACKING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** |
| SAS | Patrick Gryczka | 4 hours | 5.5 hours | 1.5 hours |
| SAS | Qiao Yang Han | 4 hours | 4.0 hours | 0.0 hours |
| SAS | Alan Huang | 4 hours | 2.5 hours | 1.5 hours |
| SAS | Kostaq Papa | 4 hours | 4.0 hours | 0.0 hours |
| SAS | Summary | 16 hours | 16 hours | 3.0 hours |

**Cumulative**

|  |  |  |  |
| --- | --- | --- | --- |
| **Who** | **Estimated** | **Actual** | **Difference** |
| Patrick Gryczka | 4 hours | 5.5 hours | 1.5 hours |
| Qiao Yang Han | 4 hours | 4.0 hours | 0.0 hours |
| Alan Huang | 4 hours | 2.5 hours | 1.5 hours |
| Kostaq Papa | 4 hours | 4.0 hours | 0.0 hours |

**12.4 DEFECT TRACING**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Artifact or Deliverable** | **Who** | **Estimated** | **Actual** | **Difference** | **Number of Defects Detected** |
| SAS | Patrick Gryczka | 5 | 4 | 1 | 7 |
| SAS | Kevin Han | 5 | 3 | 2 | 4 |
| SAS | Alan Huang | 5 | 2 | 3 | 3 |
| SAS | Kostaq Papa | 5 | 4 | 1 | 5 |
| Total Defects |  |  |  |  | 19 |

**Cumulative**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Who** | **Estimated** | **Actual** | **Difference** | **Number of Defects Detected** |
| Patrick Gryczka | 5 | 4 | 2 | 7 |
| Qiao Yang Han | 5 | 3 | 1 | 4 |
| Alan Huang | 5 | 2 | 3 | 3 |
| Kostaq Papa | 5 | 4 | 2 | 5 |
| Total Defects |  |  |  | 19 |