

## **Requirements Specification**

### **Let's Make a Match**

#### *Falcon Enterprises*

Prepared by:

Dussault, Rudolph  
Gentile, Thomas  
Mahar, Paul E  
Milton, Kimberly  
Milton, Michelle  
White, Kimberly

Clients:

Mr. Jay Bebb  
Director of Residence Life  
Siena College

Mr. Brian Peppiatt  
Assistant Director for Information Systems (Student Affairs)  
Siena College

**October 30, 2003**

**Falcon Enterprises**  
**Let's Make a Match Requirements Specification**

**Table of Contents**

Section 1: Product Overview and Summary.....	1
Section 2: Development, Operating and Maintenance Environments.....	1
Section 3: Functional Requirements.....	1
Section 4: Performance Requirements.....	3
Section 5: Exception Handling.....	3
Section 6: Early Subsets and Implementation Priorities.....	4
Section 7: Foreseeable modifications and Enhancements.....	4
Section 8: Acceptance Criteria.....	4
Section 9: Testing Requirements (requirements-based inventory).....	5
Section 10: Design Hints and Guidelines.....	6
Appendices:	
Appendix A: Glossary of Terms.....	7
Appendix B: Gantt Charts.....	9
Appendix C: Screen Mock-ups.....	11
Appendix D: Data Flow Diagrams.....	18
Appendix E: Data Elements for Data Flow.....	29

## **Section 1: Product Overview and Summary**

The Siena College Residence Life and Student Affairs Departments deals with matching hundreds of students as roommates each year. The current process used is paper based and is time consuming and produces random pairs of roommates. This process may result in many incompatible roommates pairs.

“Let’s Make a Match” is an online roommate pairing application that assists compatible roommates in their search to find each other. The application will allow roommates to create a personal profile of themselves which will then be used to pair up students with similar profiles. Once the profiles have been matched, students may contact each other by using email addresses given through the application. If two students feel they are compatible they can submit a roommate request to Residence Life and their profiles will be flagged (an indicator that they have been matched). The application also provides upperclassmen with the tools to post any vacancies they may have in townhouses and suites.

## **Section 2: Development, Operating and Maintenance Environments**

The *Let’s Make a Match* program will be developed on the Software Engineering workstations. MYSQL or Oraserv will be used as the database management system.

The users of the program include freshman students and upper class students. The system is web based, accessed through an Internet web browser. Students will be provided with a username and password when housing deposits are made. The maintenance of the database and program will be determined at a future date.

## **Section 3: Functional Requirements**

### **1. Freshman**

First freshmen must login to the system. If it is their first login, they will be asked to change their username and password. After logging in they fill out their profile. Profile questions will serve as criteria for matching. After filling out their profile completely, the students submit their profile to the system. Their information is then stored on the database and able to be searched by the queries. The students can then submit a request to receive names and contact information of students whose criteria matches theirs and what answers matched. Students can choose to contact the matches and discuss if they would like to live with each other. If they choose to live with each other, both students submit a request to the system that they wish to room together. After the system receives and confirms their request, the system flags their profile so they will not be able to be matched to other students. After every submission, confirmation is given to the student.

The following is necessary for the above set of steps to take place.

- An Online form to log in.

- Online form for freshman to create/update their profile information.
- Online form for the first time they register prompt to enter a new password. an e-mail is sent to the user if they forget their password.
- Online form to submit request to get matches.
- Online form to receive matches
- Online form to submit their living request.
- Confirm to the user after each on-line submission of information.

## 2. Returning Students

Returning students must login to the system. After login they fill out their profile. Profile questions will ask where the student wishes to live and if they are looking for someone to fill a vacancy. After filling out their profile completely, the students submit their profile to the system. Their information is then stored on the database and able to be searched. Students can then submit a request to receive names and contact information of students whose criteria matched theirs. Students can choose to contact the matches and discuss if they would like to live with each other. If they choose to live with each other, both students submit a request to the system that they wish to room together. After the system receives and confirms their request, the system flags their profile so they will not be able to be matched to other students. After every submission, confirmation is given to the student.

The following is necessary for the above set of steps to take place.

- An Online form to log in.
- Online form to enter user name and password.
- Online form to enter vacancies.
- Online to request certain living arrangements.
- Online form to receive matches
- Confirm to the user after each on-line submission of information.

## 3. Administrator

The administrator must login to the system. After logging in the administrator has several capabilities. The administrator has access to the report of confirmed matches. The administrator has the ability to flag students' profiles after their match had been confirmed. The administrator also has the ability to delete any user at any time for whatever reason. The administrator also has the ability to clear the database at the end of every year.

The following is necessary for the above actions to take place.

- On-line form to log in – administrative username and password
- Reports to show confirmation of matches
- Clear out database with confirmation that all information will be deleted
- Ability to flag people who have confirmed a match
- Ability to delete a user at anytime

## Section 4: Performance Requirements

In order for *Let's Make a Match* to perform properly, there are several constraints and requirements necessary for the system.

- The system must be secure. Only authorized students may access the system and search for roommates. The system must also be secure to prevent unauthorized use of the database.
- Processing time for the queries. The more data in the database tables the longer it may take to generate results. The length and criteria of the algorithm will also affect the response time of the query. In order to make the response time less, the query needs to be efficient and must use database search indexes.
- Reliability. The administrator must monitor and maintain the system to ensure that it is running and working properly. If students are unable to access the system they may become discouraged and not use it.

## Section 5: Exception Handling

Through normal use of any software, situations come up that are unexpected or abnormal. The system will try to prevent any situations that may arise as a result of abnormal situations and have the capabilities to fix them.

- User forgets password
  - If a student forgets their password there will be a button on the login page where they can enter their answer the hint they created in the profile and the system will send them their password by email. This will prevent students from creating new accounts if they forget their password.
- User does not fill out entire form
  - If some of the necessary information is not filled out on the form (such as email or name) the user will not be able to submit the form and it will not be saved to the database.
- Computer freezes, power is lost, loss of Internet connection or connection difficulty.
  - Student filled out the profile form information and hit the submit button.

The student may be unaware for some reason that the entered data got lost between submitting it and storing it in the database. Another scenario may be if the Internet browser cannot find the next page, the student may not know if the information is stored. To assure the student that their profile form has been completed, the student will receive a confirmation page after their information has been saved. If they do not receive a confirmation page, their profile has not been saved.

## **Section 6: Early Subsets and Implementation Priorities**

In order to meet the requirements of the system, several pieces must be completed. The most important underlying pieces of the software system are:

1. A user friendly and easy to use web site for the students to access information and forms.
2. A secure system for the data to be stored and valid user login and access to the data.
3. An online database to store all the information submitted by the students
4. Profile form for students to fill out containing matching criteria
5. An algorithm to match up students and let students know who their matches are and their contact information.

## **Section 7: Foreseeable modifications and Enhancements**

Throughout development, new ideas and situations may arise that can improve and enhance the software. Some enhancements and modifications that may happen in the future are:

- After students are matched, the database could place students in rooms on campus instead of the student affairs staff.
- Student affairs could supply a database table of all students who are going to live on campus. This table could be matched to the table of all students who created a profile in the software. The student's who have not been placed in a room, found a match or signed up for *Let's Make a Match* will be sent an email reminding them that the service is available if they wish to use it.

## **Section 8: Acceptance Criteria**

The system provides different features and access to different types of users. While a user is on-line the system allows:

1. Freshmen:
  - Choose the "looking for a new roommate" option on the *Let's Make a Match* Page.
  - Log-in to the system
  - Request forgotten password delivered by email.
  - Fill out and update profile containing matching criteria.
  - Submit their profile to the database.
  - Receive confirmation that their information had been saved in the database.
  - Request to get a matching report.
  - After finding a match, submit their match proposal to student affairs.
  - Removal of profiles if match has been confirmed with housing department.
  - Submit questions to the housing department.

- Read about housing at Siena and Siena Life policies.
2. Returning students:
- Log in to the system.
  - Request forgotten password delivered by email.
  - Fill out and update profile/form for their vacancy or living preference.
  - Submit their vacancy/living preference to the database.
  - Receive confirmation that their information had been saved in the database.
  - Submit a request to find other students who are interested in finding a room vacancy or same living preference.
  - Remove posted housing vacancy and preferences from the database to be searched.
  - Submit questions to the housing department.
  - Read about housing at Siena and Siena Life policies.
3. Administrator
- Login to special administrator account.
  - Clear all entries after housing has been assigned for the year
  - Clean up database
  - Answer questions students may have.
  - Have access to matched roommate table.
  - Submit, by email or other means, to the students' confirmation that roommate requests have been accepted.

## **Section 9: Testing Requirements (requirements-based inventory)**

### **Requirements:**

- **Students are able to log-in**  
Volunteers will be assigned a logon, sign-in and make sure they only have access to the areas granted. Volunteers will also login again to see if their information is still stored in their profile. In this stage we will also test the “lost password” option, where the users password is sent to them by email.
- **Student can create profile and update and change information in it. The information is stored in a database.**  
After logging in, the students will fill out their profile and try to update it several times. This will indicate whether the information is being saved in the database properly.

- **Student can submit a request for a roommate match and receive a report of possible matches.**  
Volunteers will fill out their profiles so that some are very close and some are very different. After filling out their profiles they will submit query request to find a match. The results will be compared to similar and different profiles. This will indicate how accurate and efficient the matching algorithm is.
- **Administrator can clear data**  
Every school year the system needs to start over with new students. The Administrator is in charge of this task. To test this we will log in to the administrator account and clear the student data tables.
- **User friendly and easy to use**  
Volunteers will be asked to provide feedback after using the system, paying close attention to parts they had difficulty with or did not understand clearly.
- **Submit a match**  
Two volunteers will select a potential match and submit their proposal. Administrator will go to the database and see if the two students are together in the matched table.

## Section 10: Design Hints and Guidelines

The *Let's Make a Match* package is an online matching system. The system matches students by the criteria they fill out in their profile. The system needs to be user friendly and easy to use. The students sign-in to the system and can change or fill out their profile. They can submit a request to generate possible matches. After they review the matches, students may communicate with the other students and decide if they would like to live together for the next semester. If they decide to become roommates they can submit their request to housing and housing will place them in an appropriate room.



## Appendix A: Glossary of Terms

**Access:** A relational database running under Microsoft Windows.

**Browser** - A browser is an application program that provides a way to look at and interact with all the information on the World Wide Web.

**Code:** A system of symbols and rules used to represent instructions to a computer.

**Controlled Decentralized (CD):** A software engineering team that has a defined leader who is in charge of the group, each of which has responsibility of other specific tasks. Overall we solve problems as a whole.

**Database** - A collection of data arranged for ease and speed of search and retrieval.

**Data flow diagram** - A representation of the functional decomposition of a system.

**Gantt Chart** - A chart that depicts progress in relation to time, often used in planning and tracking a project.

**HTML (Hypertext Markup Language):** A markup language used to structure text and multimedia documents and to set up hypertext links between documents, used extensively on the World Wide Web.

**Internet** - An interconnected system of networks that connects computers around the world via the TCP/IP protocol.

**JavaScript:** A trademark used for a programming language designed to develop applications, especially ones for the Internet that can operate on different platforms.

**Linear Sequential Model / Classic Waterfall Model** – A systematic, sequential approach to software development that begins at the system level and progresses through analysis, design, coding, testing, and support.

**MYSQL** - Open Source Database Software

**Oracle:** A relational database management system that runs on most mainframe, micro, and personal computers.

**Process** – An activity that changes or manipulates data.

**Process Model** – A software engineering strategy that is used to develop a software system.

**Query** – A data retrieval request.

**Relational Database:** A database system in which any database file can be a component of more than one of the database's tables.

**Software** - Written programs, procedures or rules and associated documentation pertaining to the operation of a computer system and that are stored in read/write memory.

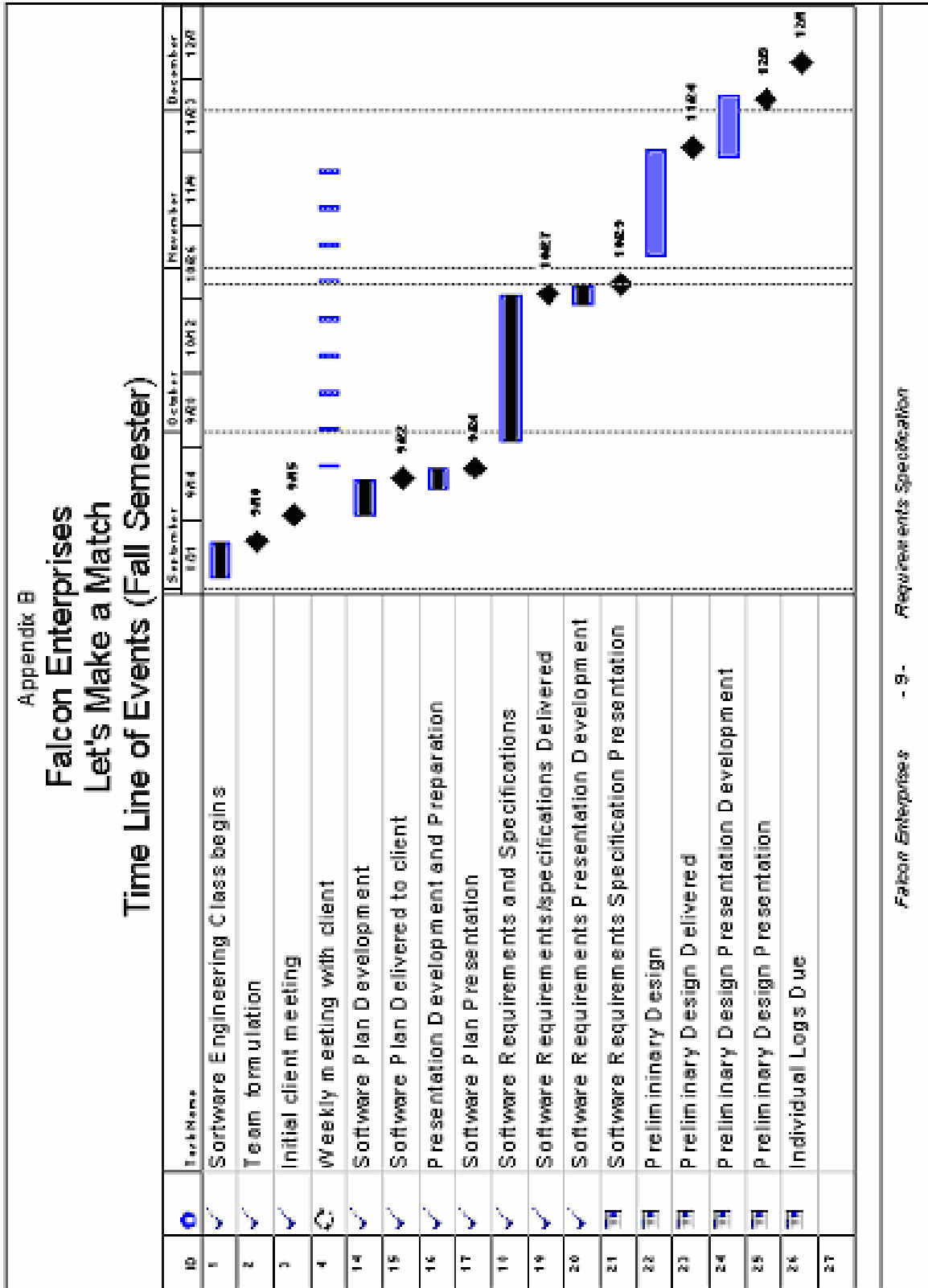
**SQL** - An industry-standard language for creating, updating and, querying relational database management systems.

**Use Case** – Set of scenarios that show a usage of the system by a certain user.

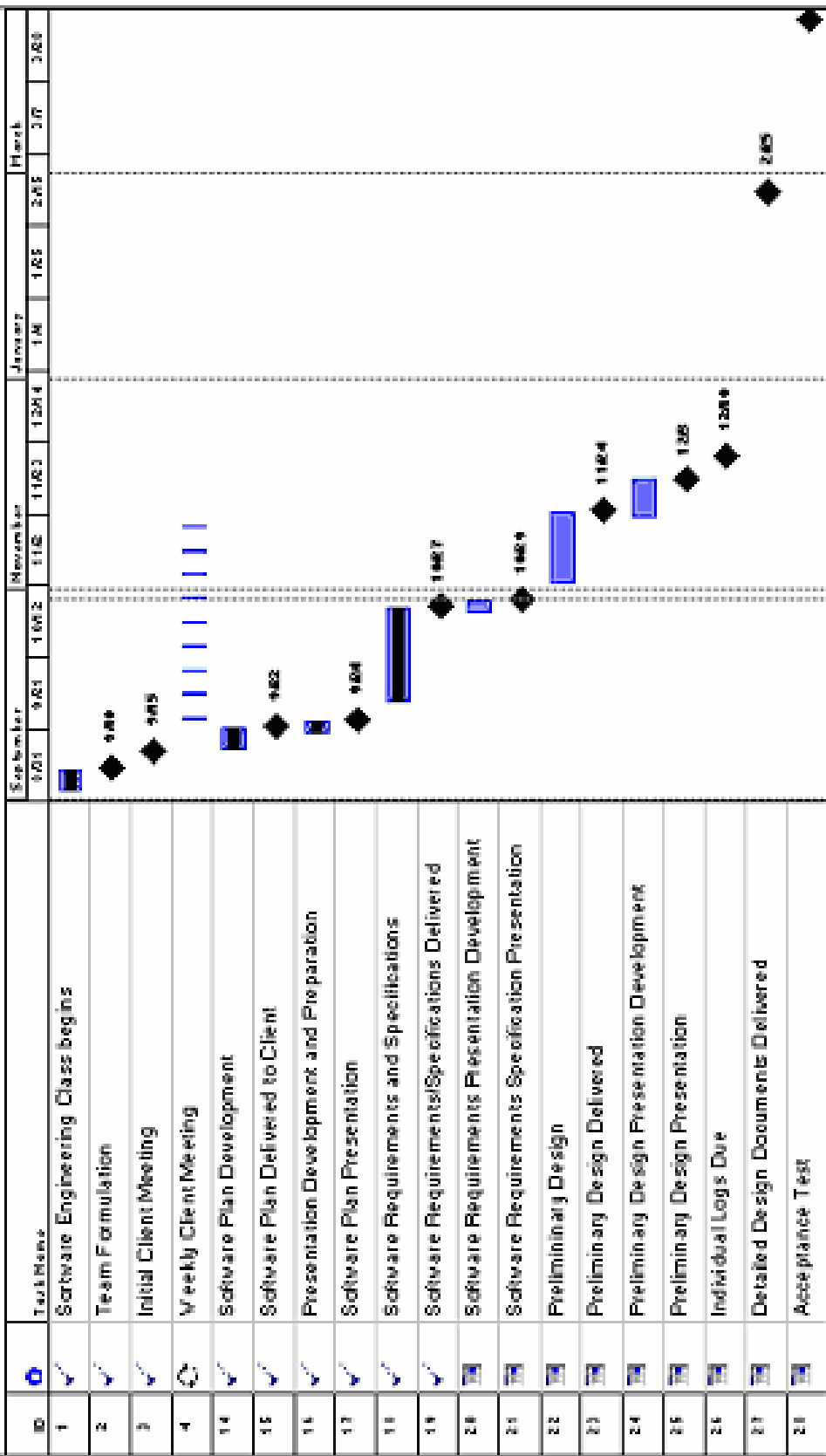
**User** – An individual that has signed on to the *Let's Make a Match* System and has been assigned a user name and password.

**Visual Basic:** A popular event-driven visual programming system from Microsoft Corporation for Microsoft Windows.

## Appendix B: Gantt Charts

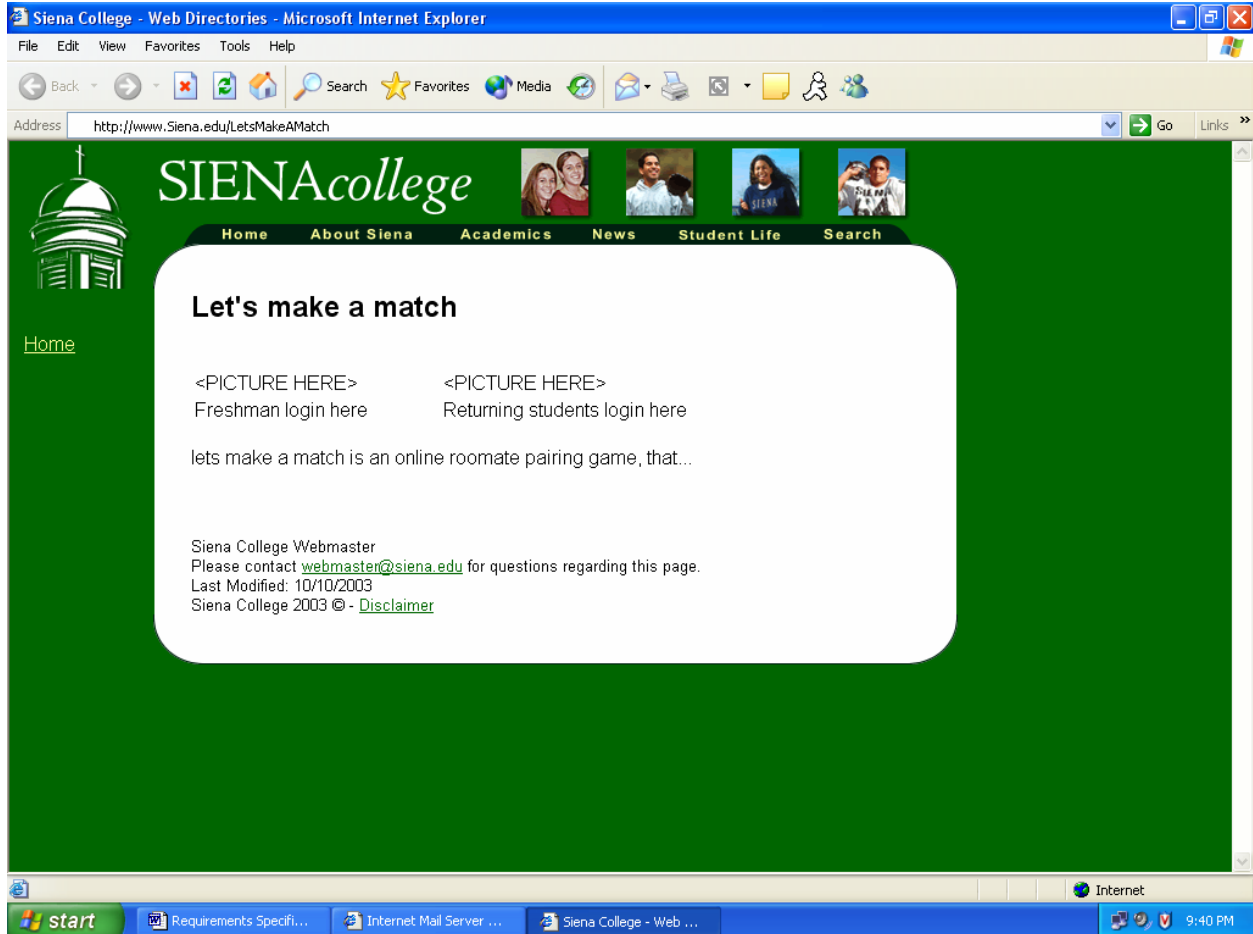


Appendix B  
**Falcon Enterprises**  
**Let's Make a Match**  
**Time Line of Events (Fall and Spring Semesters)**



## Appendix C: Screen Mock-ups

### Let's Make a Match Home Page

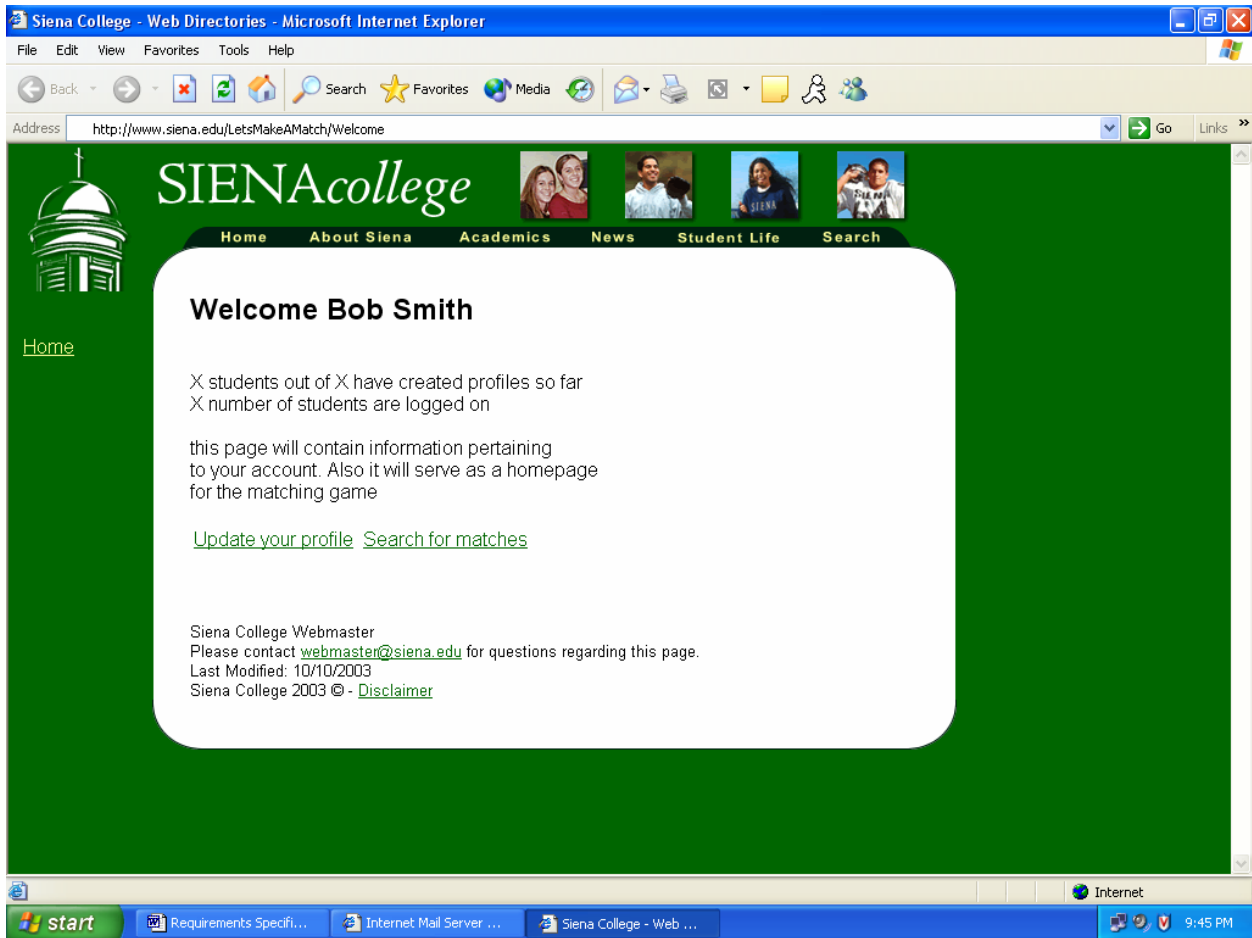


Log in Dialog box



The image shows a Windows-style dialog box titled "Connect to match.siena.edu". The title bar includes a question mark icon and a close button (X). The dialog box has a blue header area with a key icon on the left. Below the header, the text "match.siena.edu" is displayed. The "User name:" field contains a dropdown menu with a person icon and a small downward arrow. The "Password:" field is an empty text box. Below the password field is a checkbox labeled "Remember my password". At the bottom of the dialog box are two buttons: "OK" and "Cancel".

Personalized Welcome Screen/ Home Page. Users branch out into other activities from this screen.



Create/Update Profile Screen. User answers questions and the information is stored in the Database.

**Siena College - Web Directories - Microsoft Internet Explorer**

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites Media

Address <http://www.Siena.edu/LetsMakeAMatch/Profile>

**SIENAcollege**

Home About Siena Academics News Student Life Search

### Create your Profile

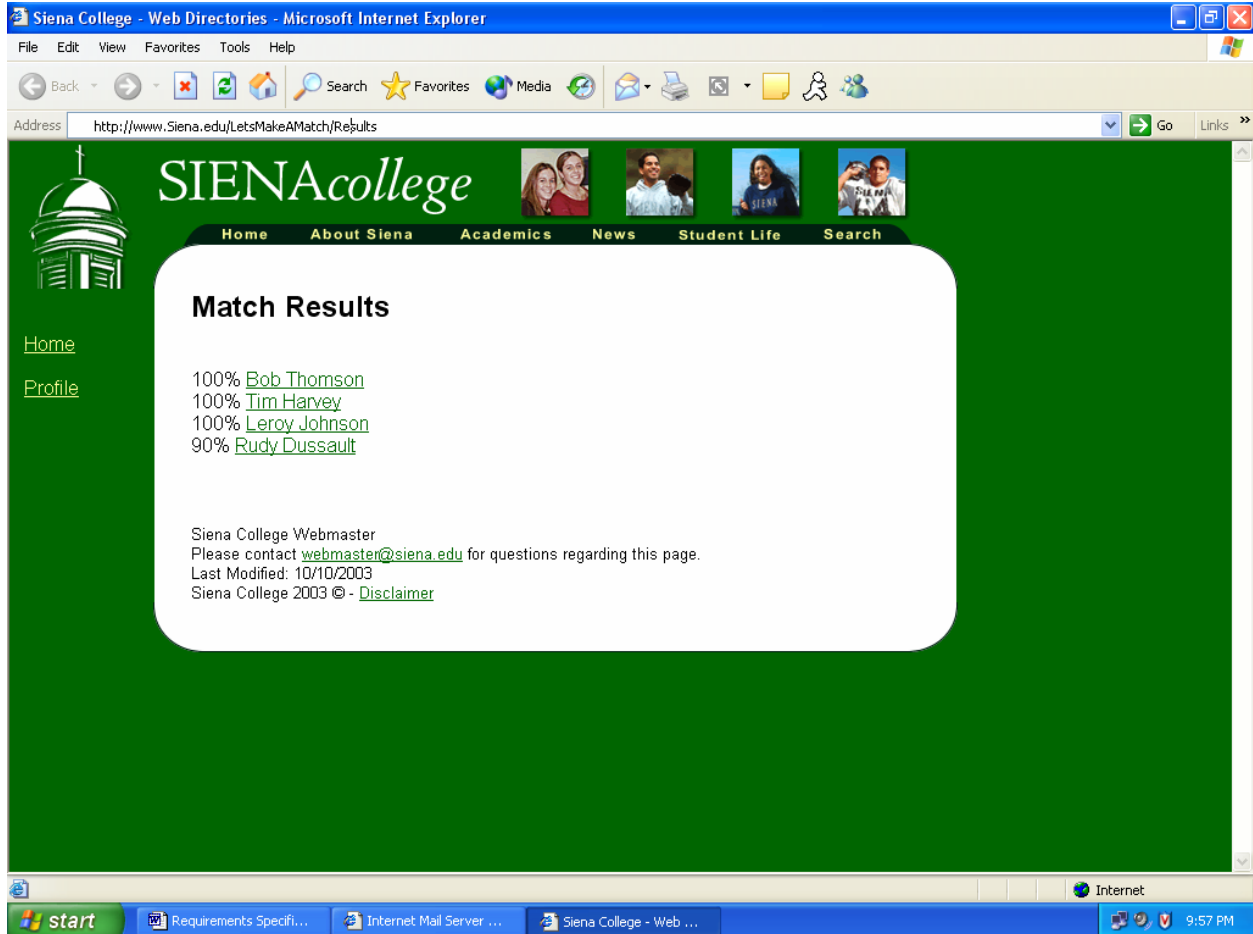
1. Do you smoke?
2. What is your degree of athletic involvement?
3. What are your sleeping habits?
4. My Radio is Usually Playing:
5. What are you study habits?  
  
Study in room if quiet during the day  
Study late night  
Probably not studying a lot over the semester
6. What school are you in at Siena?
7. How often do you drink?
8. What are some hobbies/activities that you enjoy?  
 Hiking  Reading  Singing  Soccer
9. Additional information about yourself:

Siena College Webmaster  
Please contact [webmaster@siena.edu](mailto:webmaster@siena.edu) for questions regarding this page.  
Last Modified: 10/10/2003  
Siena College 2003 © - [Disclaimer](#)

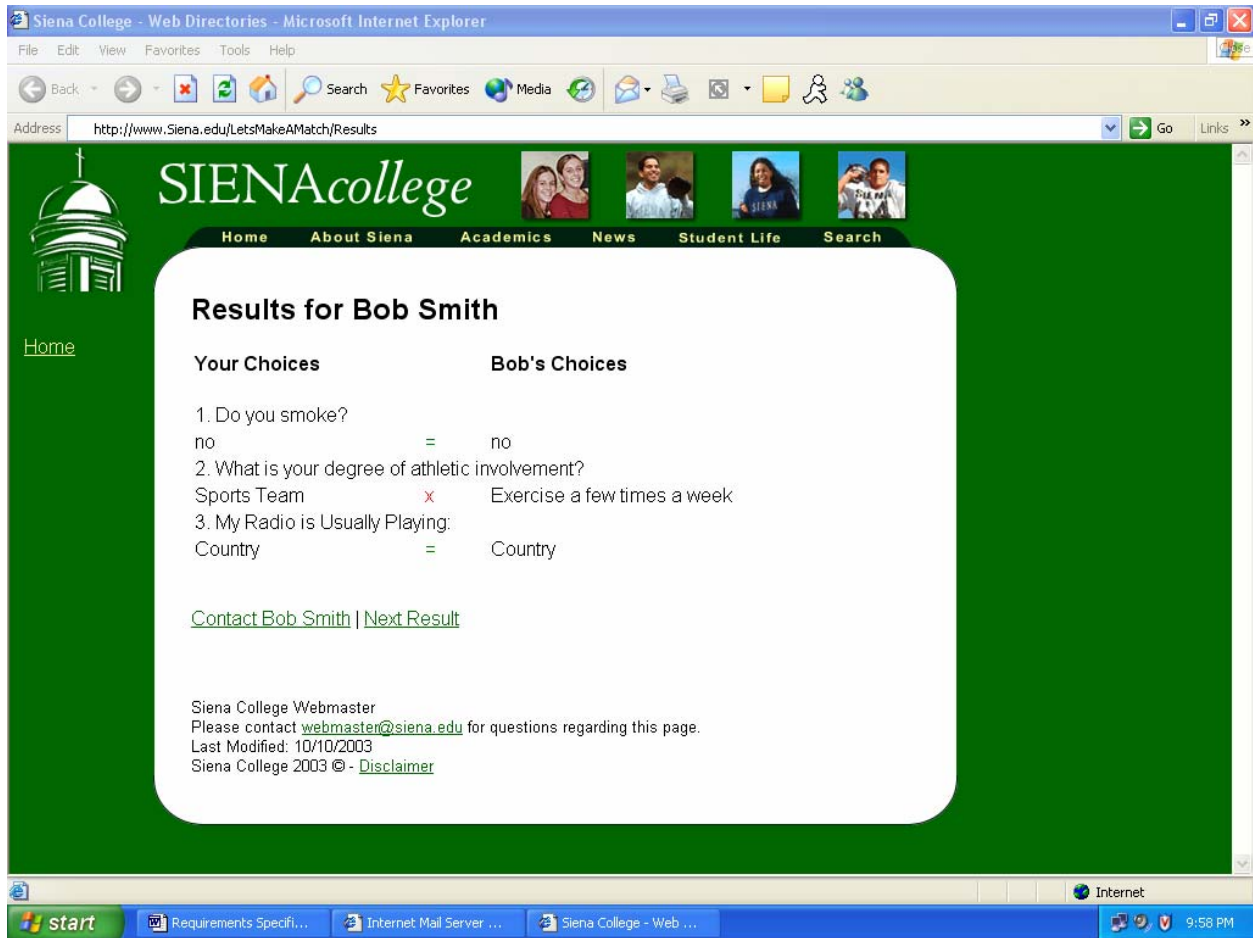
start Requirements Speci... Internet Mail Server ... Siena College - Web ... Internet



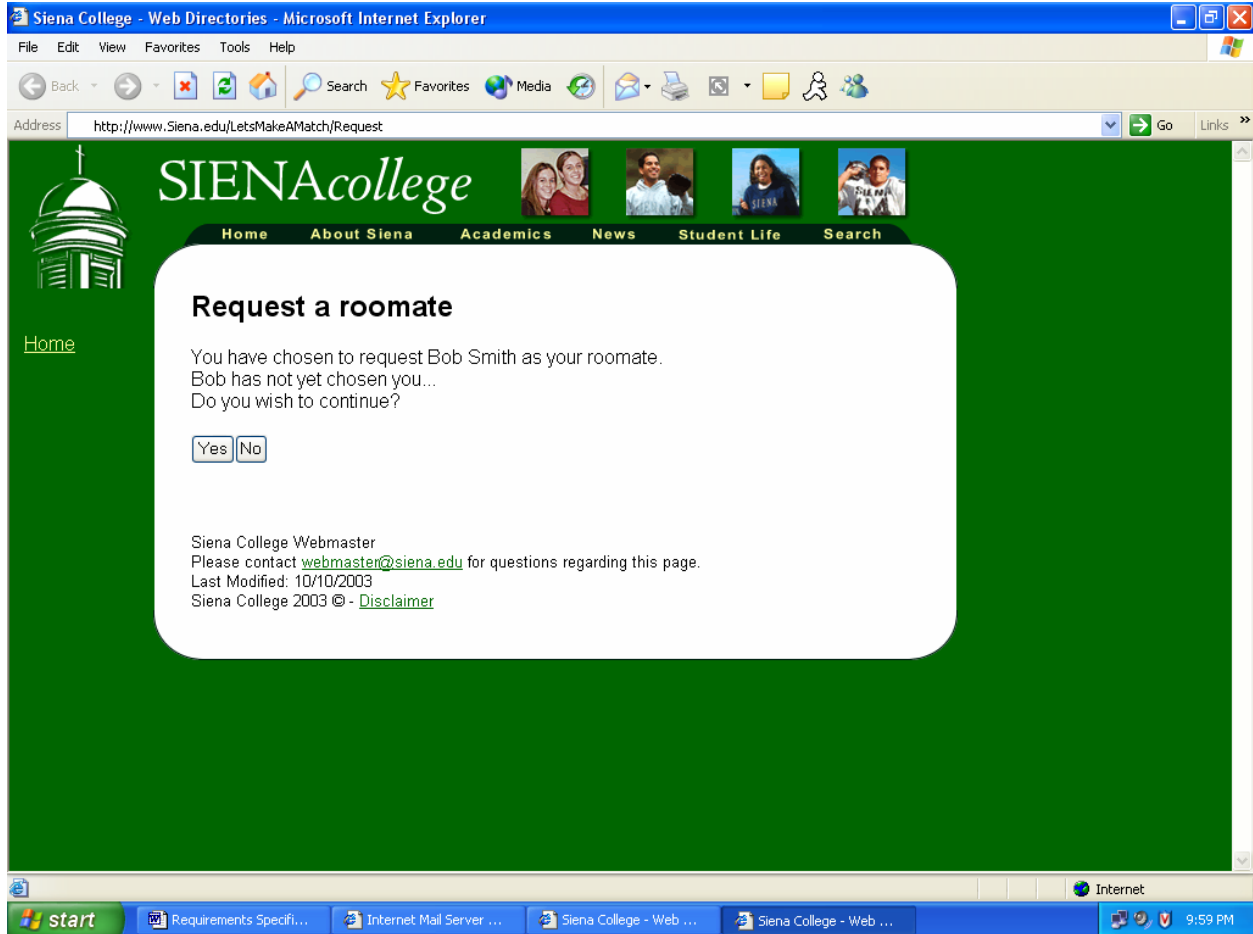
Possible Roommate Match screen. This is the results of query listed in order of relevance.



Possible Roommate information screen. You can choose to contact the potential roommate from this screen.

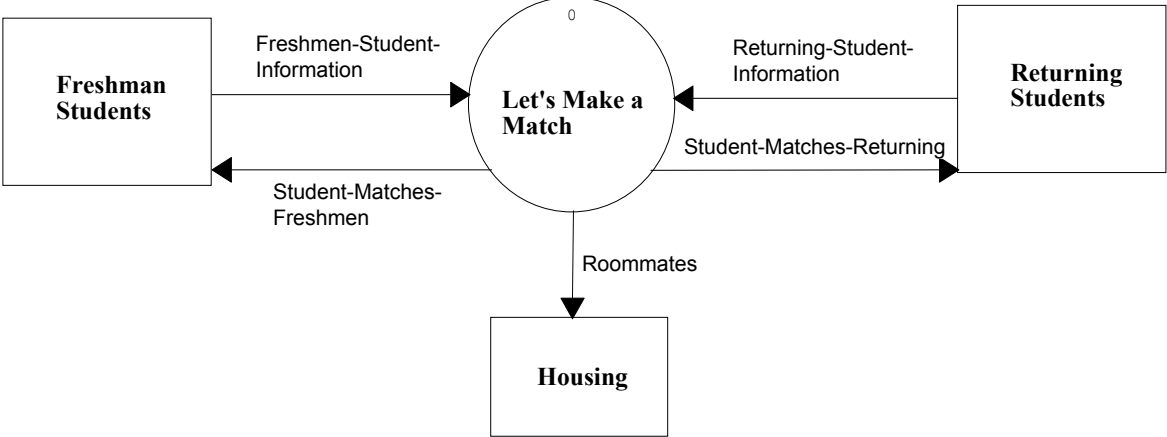


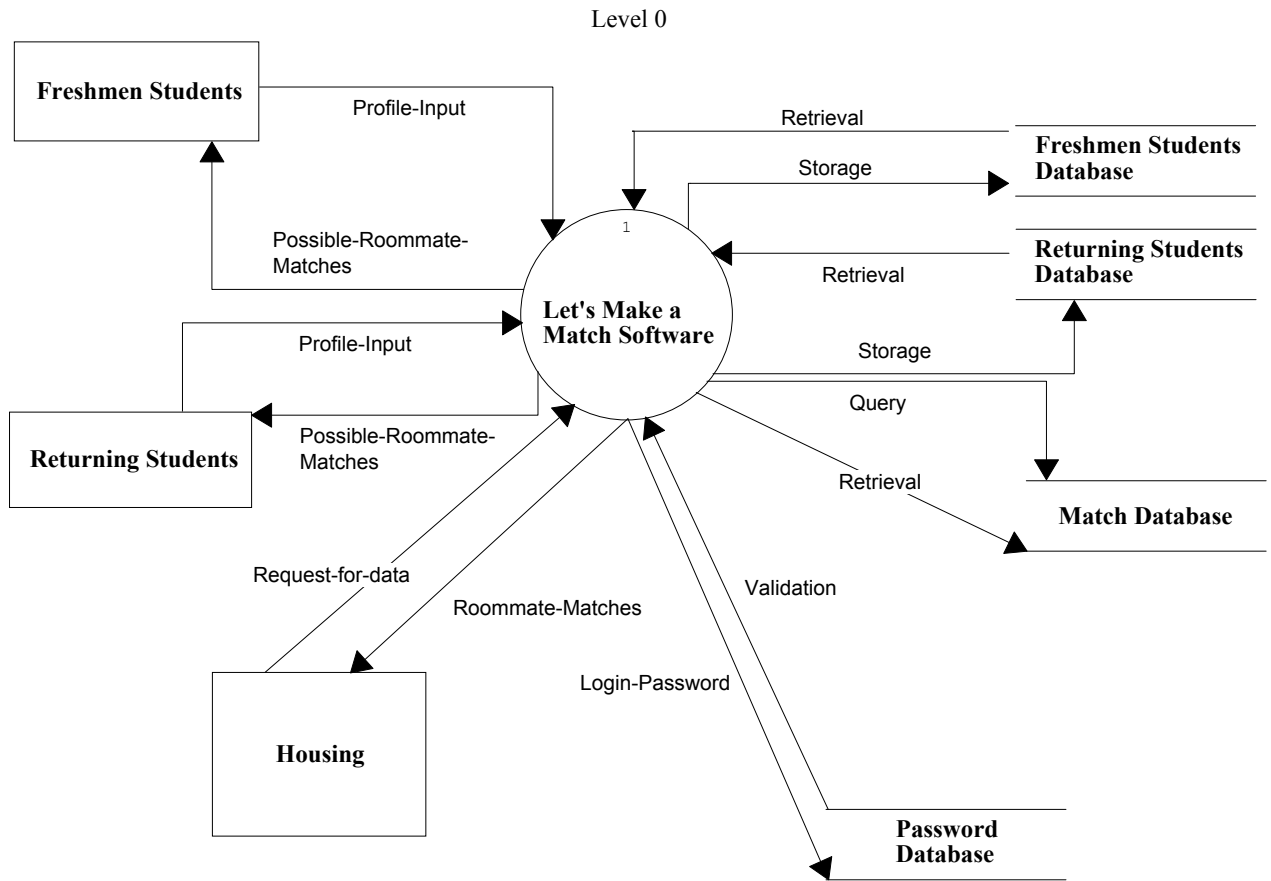
After choosing a roommate, submit your request to housing.



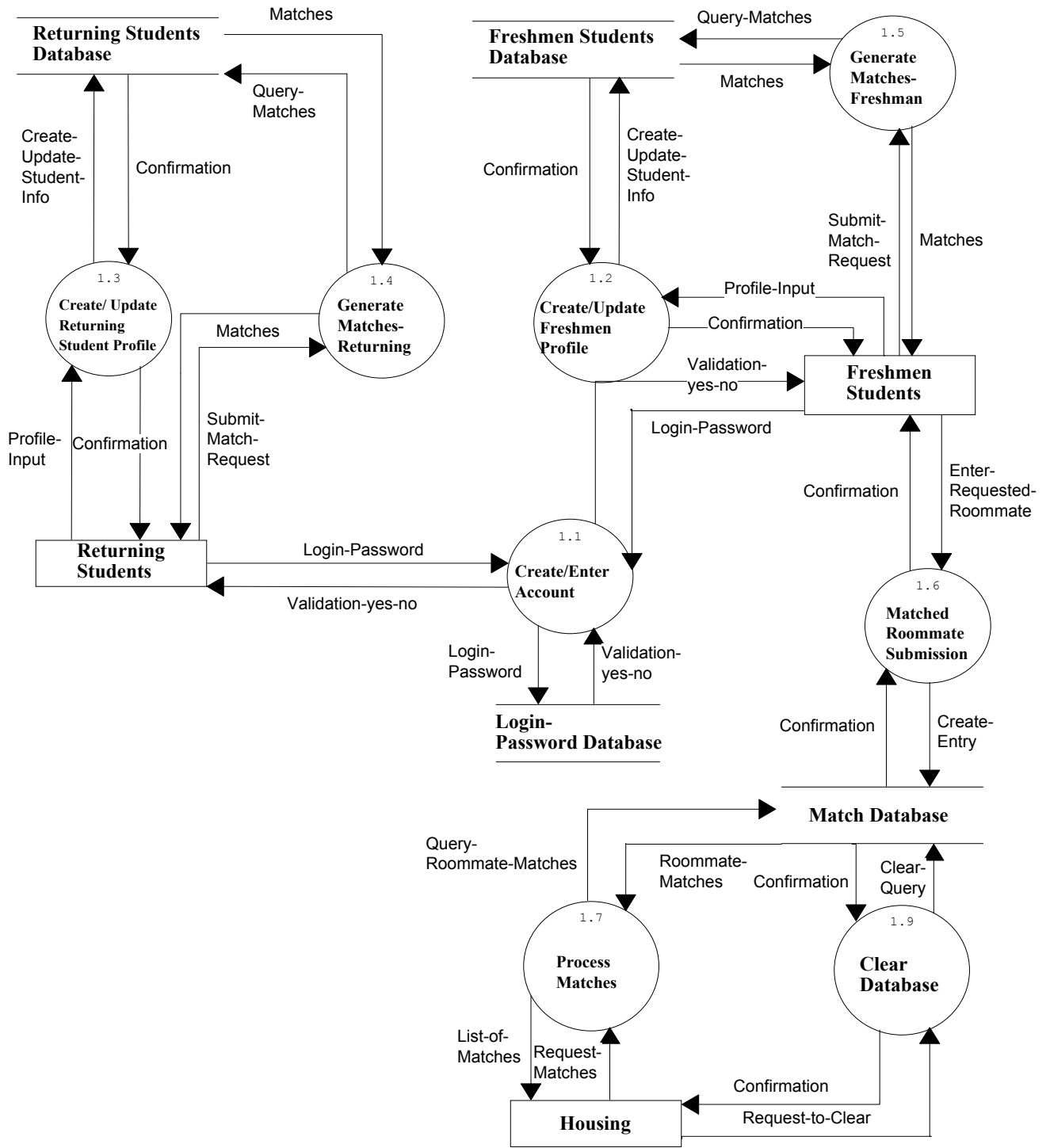
**Appendix D: Data Flow Diagrams**

Let's Make a Match Context Diagram

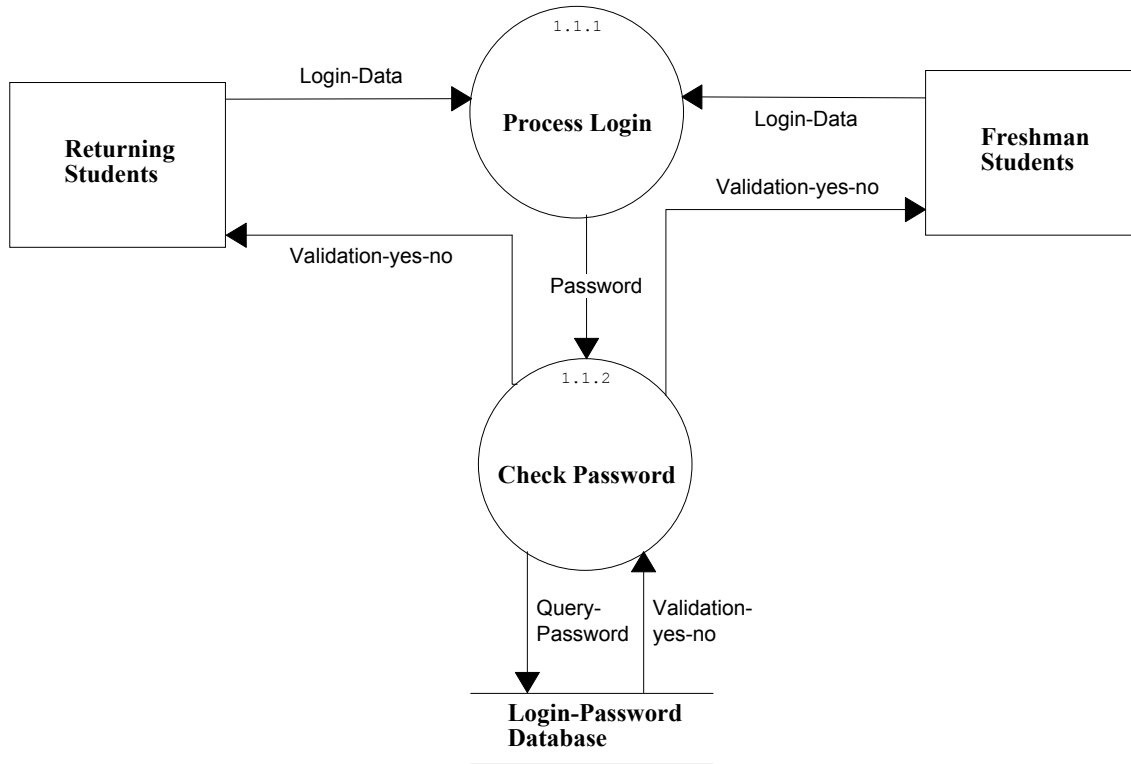




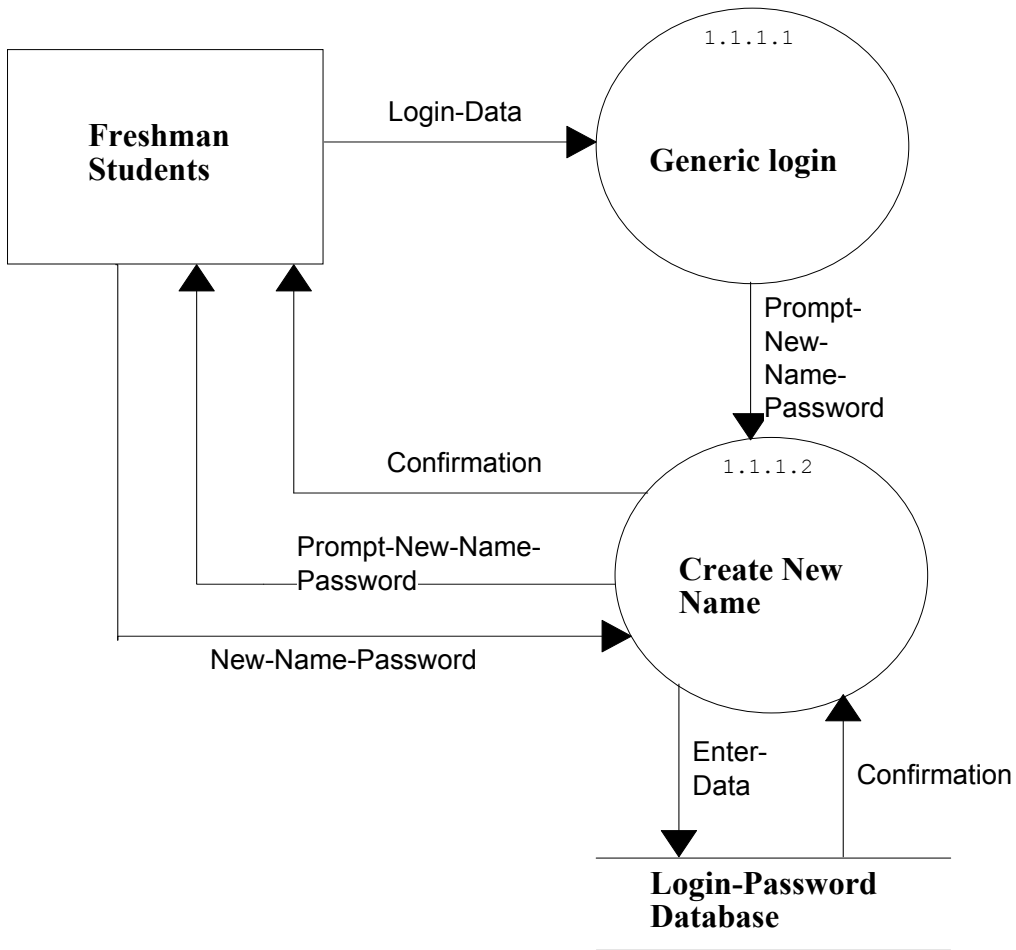
Level 1



### Create/Enter Account

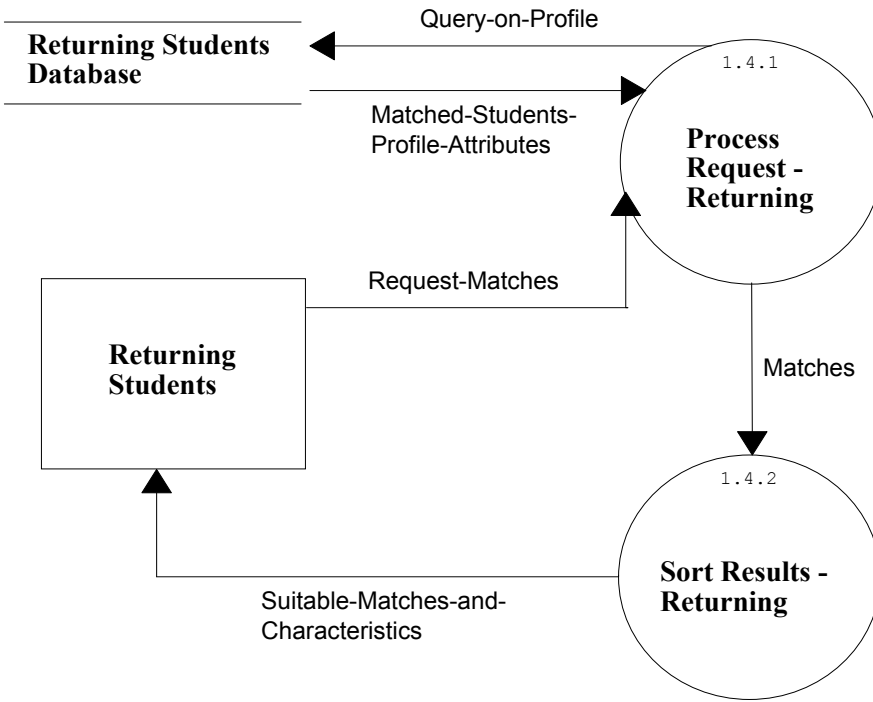


## Process Login

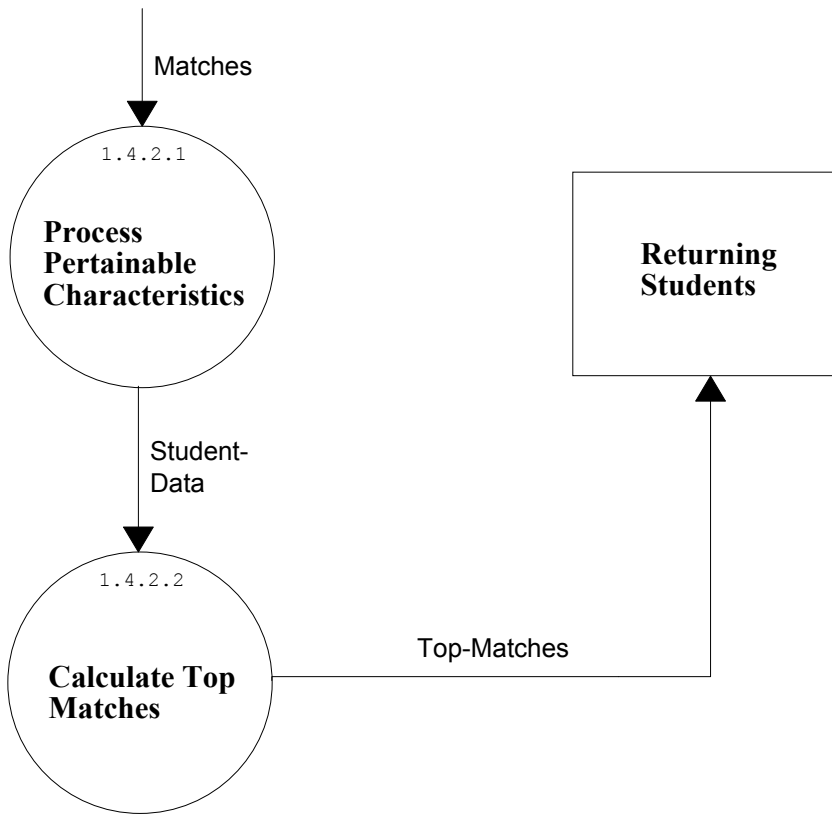




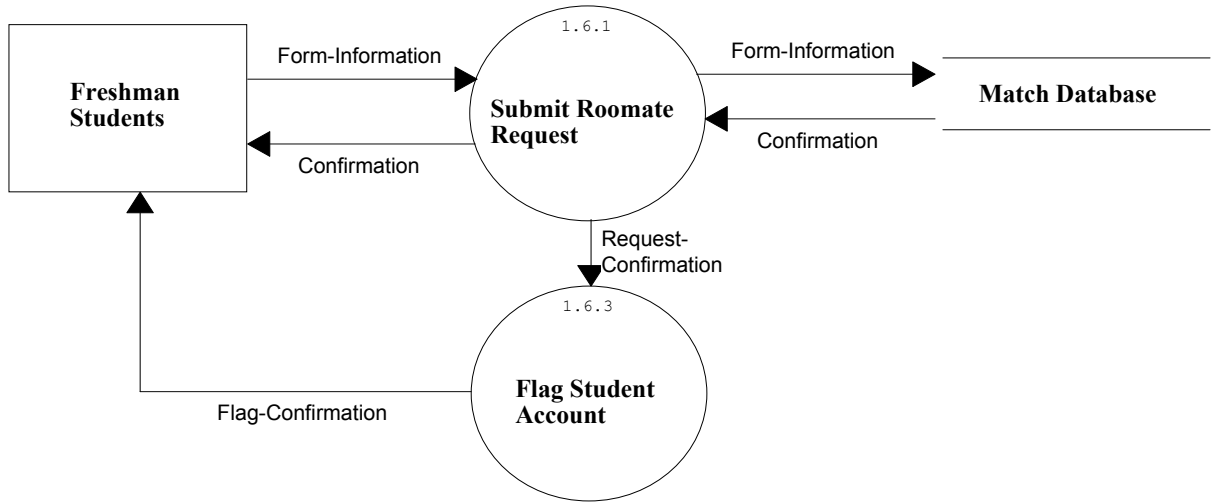
### Generate New Matches Returning



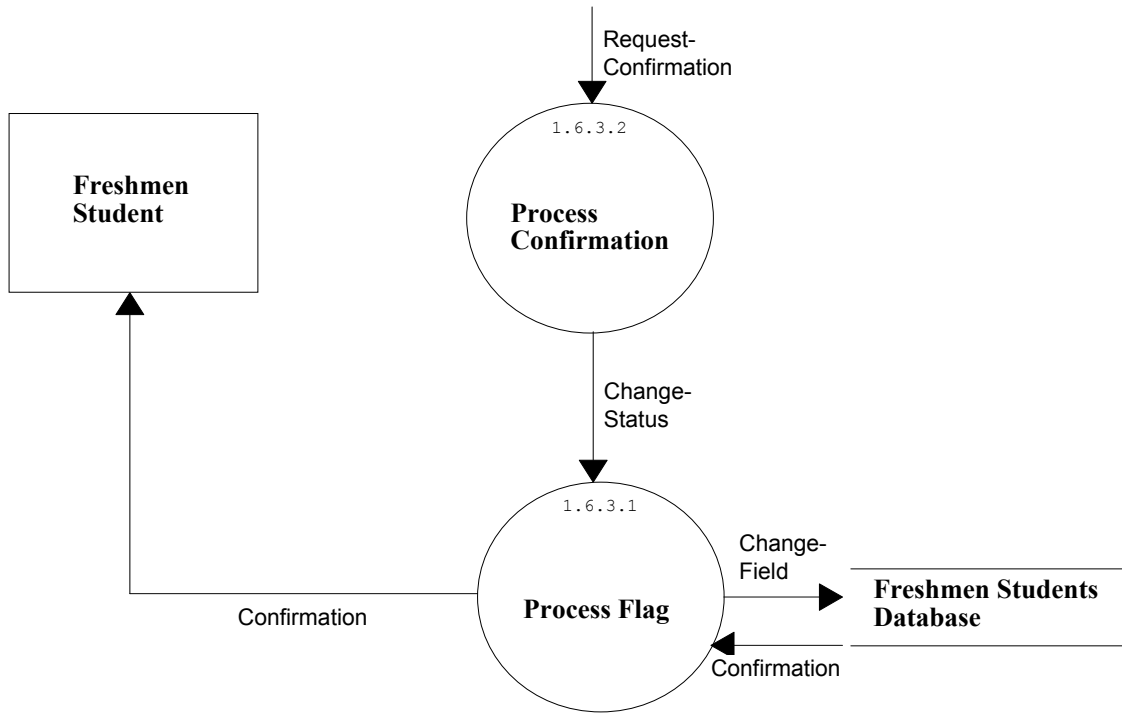
### Sort Results Returning Students



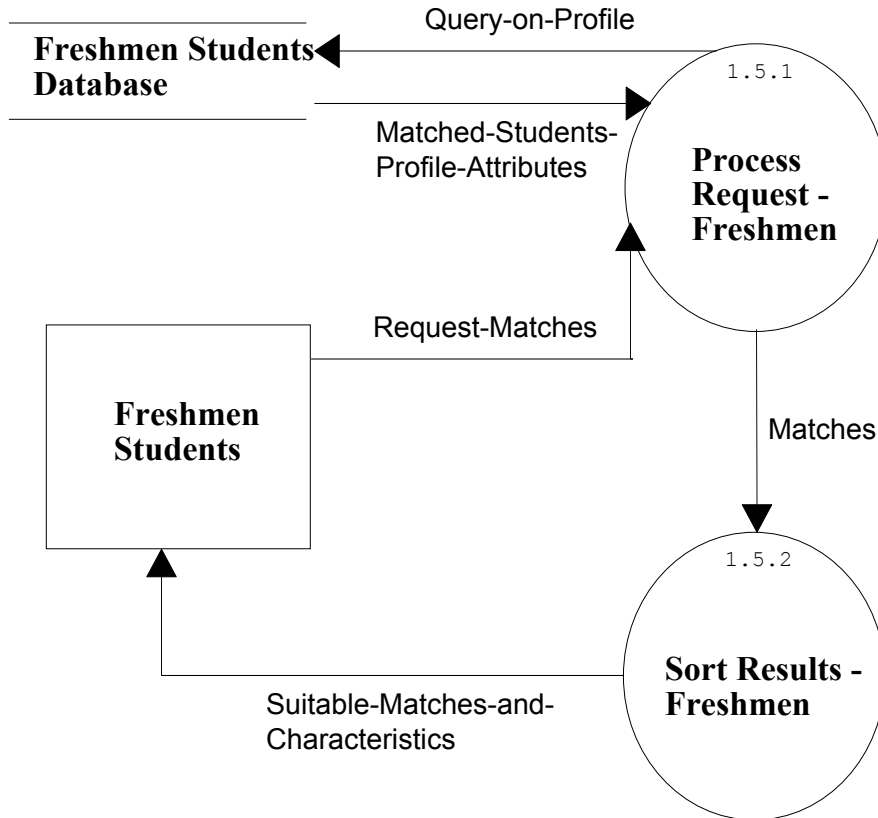
Matched Roommate Submission



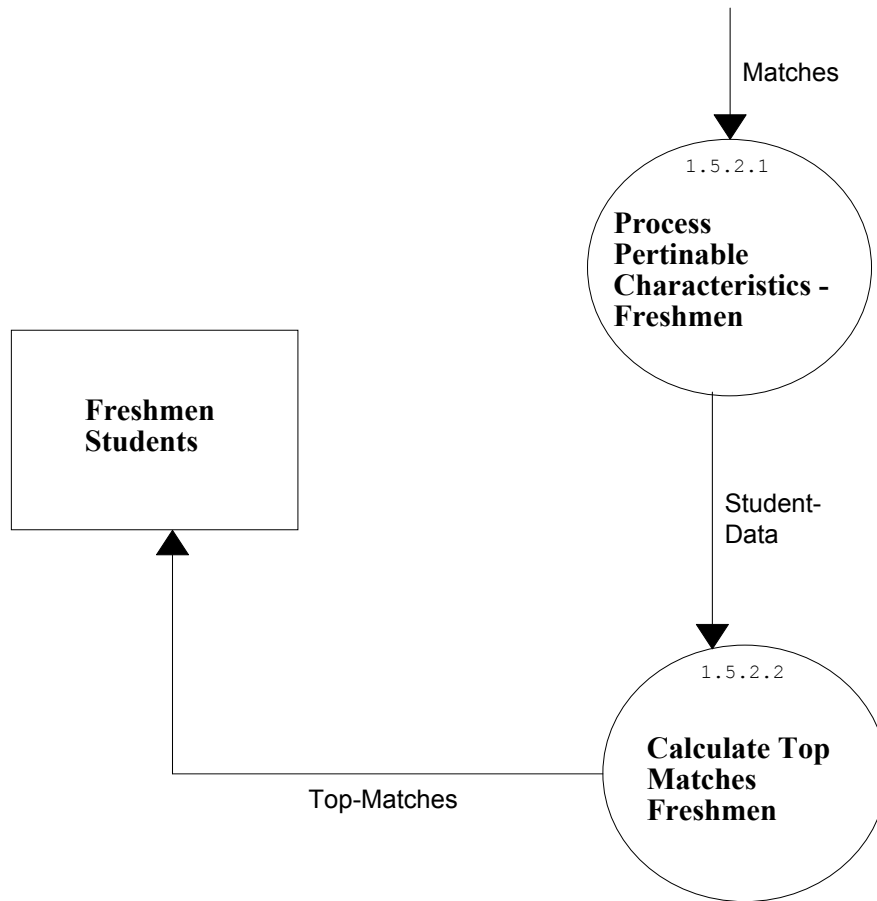
### Flag Students Account



## Generate Matches - Freshmen



## Sort Results Freshmen



## Appendix E: Data Elements

### Data Flow

#### Context Diagram

Freshmen Student Information

Returning Student Information

Student Matches Freshmen

Student Matches Returning

Roommates

#### Level 0

Login-Password

Validation

Profile Input

Possible Roommate Matches

Storage

Retrieval

### Data Elements

Elements chosen from the pull down menu, on the user profile form

Elements chosen from the pull down menu, on the user profile form

List of possible roommates and some information about them

List of possible roommates and some information about them

List of names of students that requested each other

Login and password

Yes or no

Elements chosen from the pull down menu, on the user profile form

List of possible roommates and some information about them

Elements chosen from the pull down menu, on the user profile form

SQL Code that returns a list of possible roommates and some information about them

(The data types are the same for both freshmen and returning students, although the actual information returned may be different.)

Request for Data	Button clicked that signals a database query (code)
Query	SQL code
Retrieval (from Match Database)	List of names of students that requested each other
Roommate Matches	List of names of students that requested each other
<b><u>Level 1</u></b>	
Login Password	User name and password
Validation	Yes or No
Profile Input	Elements chosen from the pull down menu, on the user profile form
Create Update Student Info	Elements chosen from the pull down menu, on the user profile form
Confirmation	Success or Failure
Submit Match Request	Button clicked that invokes a query (code)
Query Matches	SQL Code
Matches	List of possible roommates and some information about them
(The data types are the same for both freshmen and returning students, although the actual information returned may be different.)	
Confirmation	Success or Failure
Enter Requested Roommate	User Input (name)
Create Entry	SQL Code
Request Matches	Button clicks that invokes a Query (code)



Query Roommate Matches	SQL Code
Roommate Matches	List of names of students that requested each other
List of Matches	List of names of students that requested each other
Request to Clear	Button clicks that invokes a Query
Clear Query	SQL Code

### **Create Enter Account**

Login Data	Login Name (Screen Name)
Validation	Yes or No
Password	Password
Query Password	SQL Code

### **Process Login**

Login Data	Login Name (Screen Name)
Prompt New Name Password	A new window appears to enter data (code)
Enter Data	New Login name (Screen Name)
Confirmation	Acceptance or Failure
New Name Password	Login and Password

### **Generate New Matches Returning / Generate New Matches Returning**

Request Matches	Button clicked to signal database (code)
Query on Profile	SQL Code

Matched Students Profile Attributes	Raw list of attributes on potential roommates
Matches	Raw list of attributes on potential roommates
Suitable Matches and Characteristics	List of possible roommates and some information about them

### **Sort Results Returning Students / Sort Results Freshmen Students**

Matches	Raw list of attributes on potential roommates
Student Data	Modified list of possible roommate matches with attributes taken out that students can not see about each other
Top Matches	List of possible roommates and some information about them

### **Matched Roommate Submission**

Form Information	Requested roommates name
Confirmation	Acceptance or Failure
Request Confirmation	Acceptance
Flag Confirmation	Acceptance or Failure

### **Flag Students Accounts**

Request Confirmation	Acceptance
Change Status	Change variable STATUS to NO
Change Field	SQL Code
Confirmation	Acceptance or Failure