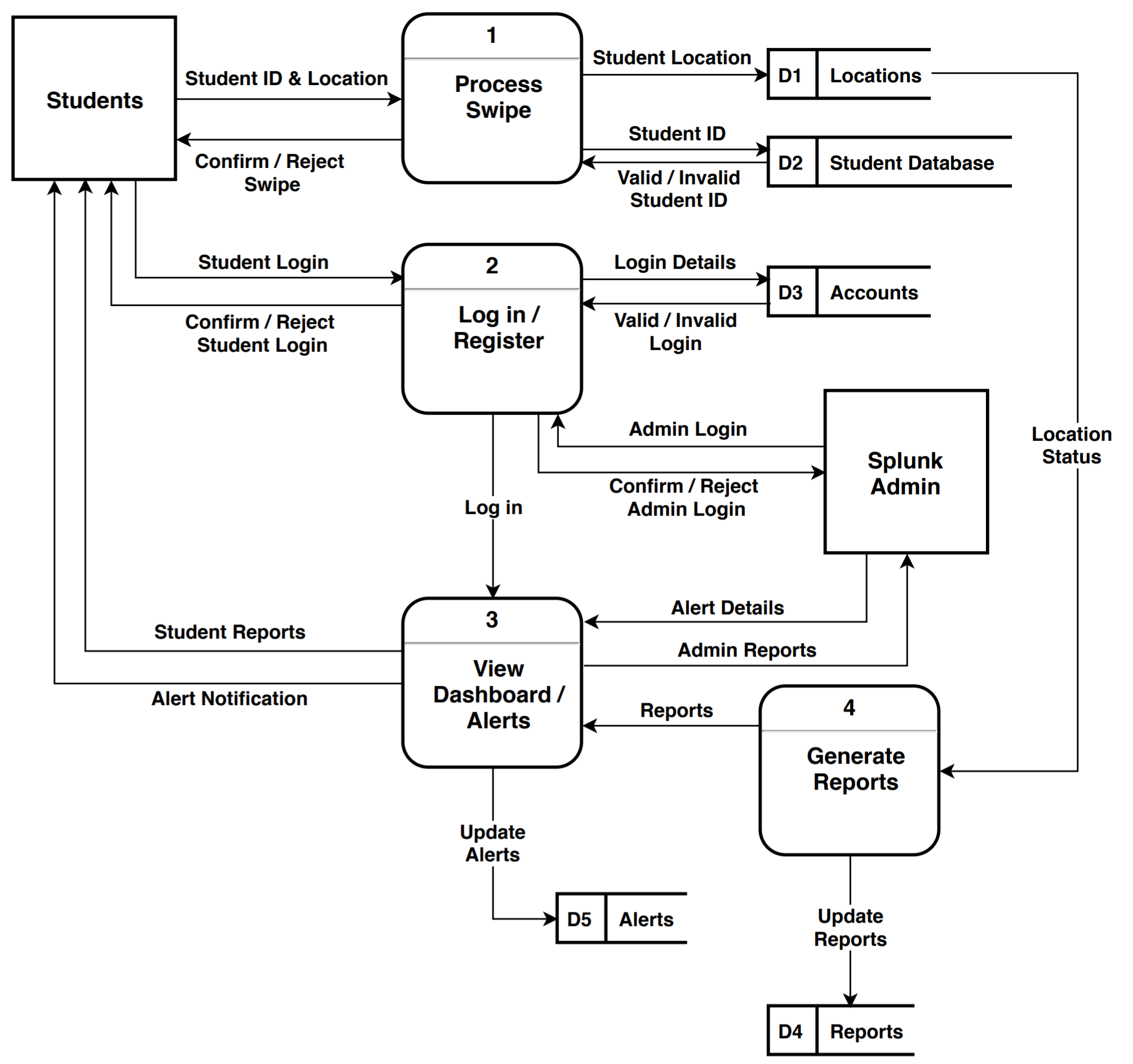
****

**Processes & Data Flows:**

* Process Swipe:
  + **Number:** 1
  + **Name:** Process Swipe
  + **Description:** This process will perform a couple of actions. First, it will validate that once a student id is swiped, they are indeed a faculty, student, or staff member of the University. If this is found to be true, the location and ID number of the individual who swiped their ID will be sent to a database where Splunk will be configured to pull data from in real-time. This would be important when analyzing traffic in areas on campus because the more swipes per specified time frame within a general vicinity, the more likely it will need to be highly staffed.
  + The **input data flow** for this process would be the student ID swipe. If this student, staff, or faculty identity is confirmed then a corresponding access control action will be initiated.
  + The **output data flows** for this process would be sending the student ID and student location information to the backend database.
* Log in/Register:
  + **Number:** 2
  + **Name:** Login/Register
  + **Description:** This process will handle the login/registering students/staff/faculty into the Splunk system to give them permissions/access to dashboards and visualizations in real-time. First, the individual will enter their login credentials into the system login screen. Those login credentials will be verified with the already existing accounts within the Splunk system. If the identity is confirmed, then access will be granted and they will be able to freely use the application within their assigned permissions/roles (if Admin role is assigned with credentials, then the corresponding dashboards/reports will be visible to the user). If the identity is unknown, the system will deny their request and ask for them to either re-enter proper credentials, or create an account using their UMBC login.
  + The **input data flows** are the login credentials entered.
  + The **output data flows** would be verifying the credentials are active in the accounts database, verifying the roles/permissions associated with the account, and displaying relevant dashboards and reports to the user.
* View Dashboard/Alerts:
  + **Number:** 3
  + **Name:** View Dashboard/Alerts
  + **Description:** Depending on the results from the previous process (login stage), certain dashboards/reports/alerts will be generated and displayed for the user. If an admin has logged in, corresponding alerts and dashboards will be available for them. If a student is logged in, then they will be able to see the dashboards they’re permitted to see, and alerts they’re able to access and receive.
  + The **input data flows** are the login credentials, and alert details from the previous process.
  + The **output data flows** include corresponding admin reports, student reports, and alert notifications. Also, if any alerts need to be updated by the user, the output will reflect the updates.
* Generate Reports:
  + **Number:** 4
  + **Name:** Generate Reports
  + **Description:** In this process, we get inputs from the location status forwarded to the indexer from the first process. These location status reports will be used to update the dashboards in real-time to reflect the most recent changes in student traffic around campus. When the reports are updated, the output will reroute back to process 3 to display these reports to the user.
  + The **input data flows** is location status.
  + The **output data flows** are updating the reports, and making sure they’re displayed in Process 3.

**Entity:**

* **Students:** The students will interact with the system using the UI on the app. They will retrieve information from the application about student traffic on campus, LMS engagement, and other statistics that they will have permissions to. The only data they will be contributing will be when they swipe their student ID’s to access different areas on campus. Without their swipes, this application will not accurately reflect the goals.
* **Splunk Admin:** The Splunk Admin’s will interact with the system on the backend, ensuring that dashboards are up to date and data flow is smooth. They will also have delete access if they would like to delete records (i.e. records indicating denied access). They could also create and edit saved searches, run searches, edit its own preferences, create and edit event types, create and edit alerts, and other similar tasks.

**Data Store:**

* **Locations:** This data store will hold all the information about the location data taken from student swipes around campus. This would make it easier for Splunk to retrieve from a specific data source, rather than parsing and indexing through a variety of data sources not specified. This will make the transaction time increasingly faster.
* **Student Database:** This data store will hold all the information about the student information (login credentials, whether current student or alum). This will be used to verify login in Process 2.
* **Accounts:** This data store will hold all the credentials about login. If the system receives a login that isn’t already existing in the database, then it will inform the user that they have entered incorrect credentials and to try again. There will also be a ‘sign up’ option where the user has the ability to create a new login. Ideally, the login credentials will be the same as their UMBC myUMBC login.
* **Alerts:** This data store will hold all the current alerts that have been set by the user to be triggered off a certain criteria. If the user would like to modify/add/delete alerts, they could do so on the UI which will send queries to the backend database to make these preferred changes.
* **Reports:** This data store will hold all the reports that will be generated upon startup, or triggered reports based on set intervals. If the user would like to modify/add/delete reports (based on their permissions as well), then they could do so.