Android Development Final Project Proposal – 3/14/17

**Team Members:** Yuri Khechoyan & Leah Perry, & Jessie Wilkins

**Project Title:** DunDeal ***(working app title)***

**Business Requirements:**

* Create a seamless request based Application that helps college students get things done
* Elevate the stress of having to complete a monumental number of tasks within 24 hours
* Assist those students that are completing the tasks ***(dundies – working title for these students)*** to make some extra cash on the side by offering monetary compensation for their diligent work.

|  |  |
| --- | --- |
| **Use Case 1: Create Account (if not already created)** | |
| **Scope** | Business |
| **Level** | Sea Level |
| **Goal in Context** | Allows the Students to create a new account with DunDeal (new customer acquisition) |
| **Primary Actor** | Student |
| **Stakeholders** | Developers, Students |
| **Preconditions** | Application is launched |
| **Triggers** | Student taps on the Create Account button |
| **Success Guarantee** | Application launches a form in which user input is required |
| **Main Success Scenario:** | |
| DunDeal account is created | |
| **Extensions:** | |
| Items needed for successful confirmation:  **1a.** Enter Full Name (First & Last)  **1b.** Enter Username – verify that username does not exist in database already  **1c.** Enter Password (initial and re-entry)  **1d.** Enter Email Address (must be a .edu extension)  **1e.** Enter Mobile Phone Number  **1f.** Enter Location (State, City, Zip Code, & Country)  **1g.** Enter name of University/College student is attending  **1h.** Enter Method of Payment (MasterCard, Visa, PayPal, Android Pay, etc.)  i. Card Number  ii. Name on Card  iii. Expiration Date  iv. 3-digit CVV security code (back of card)  If these fields have an invalid value, throw toast to let Student know the field is required – let them re-enter credentials | |

|  |  |
| --- | --- |
| **Use Case 2: Login - Username** | |
| **Scope** | System |
| **Level** | Sea Level |
| **Goal in Context** | Present EditText field to Student to enter their Username Credential |
| **Primary Actor** | Student |
| **Stakeholders** | Student, Developers |
| **Preconditions** | Application is launched |
| **Triggers** | Student taps on Username (EditText field) – virtual keyboard appears |
| **Success Guarantee** | Student successfully enters their Username correctly |
| **Main Success Scenario:** | |
| Student enters their Username correctly and moves on to enter their password | |
| **Extensions:** | |
| N/A | |

|  |  |
| --- | --- |
| **Use Case 3: Login - Password** | |
| **Scope** | System |
| **Level** | Sea Level |
| **Goal in Context** | Present EditText field to user to enter their Password Credential |
| **Primary Actor** | Student |
| **Stakeholders** | Student, Developers |
| **Preconditions** | Student enters their Username (first) |
| **Triggers** | Student taps on Password (EditText field) – virtual keyboard appears |
| **Success Guarantee** | Student successfully enters their Password correctly |
| **Main Success Scenario:** | |
| 1. Student enters all their credential correctly  2. Student taps on the "Login" button when they are finished typing in their credentials | |
| **Extensions:** | |
| **3a.** Login fails and toast is thrown if:   1. Username is incorrect or does not exist 2. Password is invalid & does not match account records | |

|  |  |
| --- | --- |
| **Use Case 4: Main Menu (Request Task to be completed)** | |
| **Scope** | System |
| **Level** | Sea Level |
| **Goal in Context** | Once logged in, student has the option to create a task that needs to be taken care of |
| **Primary Actor** | Student |
| **Stakeholders** | Student, Developers |
| **Preconditions** | Student is successfully logged in |
| **Triggers** | N/A – App lands on Request Task tab by default |
| **Success Guarantee** | Students enters: task title, selects duration in which task needs to be completed by, selects whether task is personal or academic (radio buttons), & checks box (check box) for whether the task is urgent, then submits request |
| **Main Success Scenario:** | |
| Confirmation toast is initialized that shows request was submitted | |
| **Extensions:** | |
| N/A | |

|  |  |
| --- | --- |
| **Use Case 5: Main Menu (Dundie – Find open tasks to complete)** | |
| **Scope** | System |
| **Level** | Sea Level |
| **Goal in Context** | Once logged in, student taps on the Dundie tab – student has the alternative option to complete the tasks that are currently posted & available |
| **Primary Actor** | Student |
| **Stakeholders** | Student, Developers |
| **Preconditions** | Student Logged in |
| **Triggers** | Student taps on Dundie tab |
| **Success Guarantee** | Dundie will be able to see a list of the current tasks that are posted – urgent tasks are at the very top of the page, followed by the rest of the currently opened tasks |
| **Main Success Scenario:** | |
| Dundie taps on the task that they want to complete - information is given about the task: title, duration in which it needs to be completed by, a MapView (Google Maps) of where task will need to take place, and 2 options for the Dundie to choose from: Accept or Decline task | |
| **Extensions:**  **5a.** If accepted: task is removed from overall list of tasks (within certain radius) & timer for the task starts (with a complete button below it) – if the task is not completed within the duration allowed by requester, Dundie will not get paid for the task  **5b.** If declined: task if moved to the bottom of the list and will only be removed when a Dundie completes it**\***  **\***- Urgent tasks will stay at the top of the list. These tasks will NOT move to the bottom of the list if declined | |

|  |  |
| --- | --- |
| **Use Case 6: Task Completion** | |
| **Scope** | System |
| **Level** | Sea Level |
| **Goal in Context** | Notify parties involved that task has been completed |
| **Primary Actor** | Student |
| **Stakeholders** | Student, Developers |
| **Preconditions** | Task is Accepted |
| **Triggers** | Dundie goes back to the timer screen for that task, taps the “Task Completed” button |
| **Success Guarantee** | Dundie completes task within duration given by requester |
| **Main Success Scenario:** | |
| Requester of the task will receive a notification letting them know that their task has been completed.  Dundie receives an alert box with a “cha-ching” sound effect – signaling that they have completed the task. | |
| **Extensions:**  NA | |