

**CSIS 3275 Project Proposal – Untitled**  
**The Untitled Help Desk Application (UHDA)**

Jacob Pauls  
Gregory Pohlod  
Michael Wilson

## Project Scope

**The Untitled Help Desk Application (UHDA)** is a helpdesk ticketing solution designed with IT professionals, prospective users, and an everchanging world of technology in mind. As technology continues to advance and a greater breadth of tech becomes required within an organization, even more technological problems will occur. Undoubtedly, the main goal of any helpdesk/IT department should be to provide effective and proactive service that increases problem management and enables the deployment of technological infrastructure across an organization. This product serves as a means for an IT department to increase and enable complete focus on the issues that they think matter the most. It keeps simplicity in mind and considers the fact that today's IT departments are incredibly swamped with user requests from a varying number of communication platforms. With **UHDA**, the product focus will be on the user experience; from both sides of the helpdesk.

Users will easily be able to open tickets that pass into the IT department's queue. IT professionals can immediately focus their request queue by resolution time, priority, and issue type. In the application, IT can asynchronously comment, question, or update the status of a ticket so that users are always up to date on the status of their request. With the Slack integration, users can direct message the professionals in charge of their ticket (or visa versa) and more viably communicate about technological issues or questions related to the ticket.

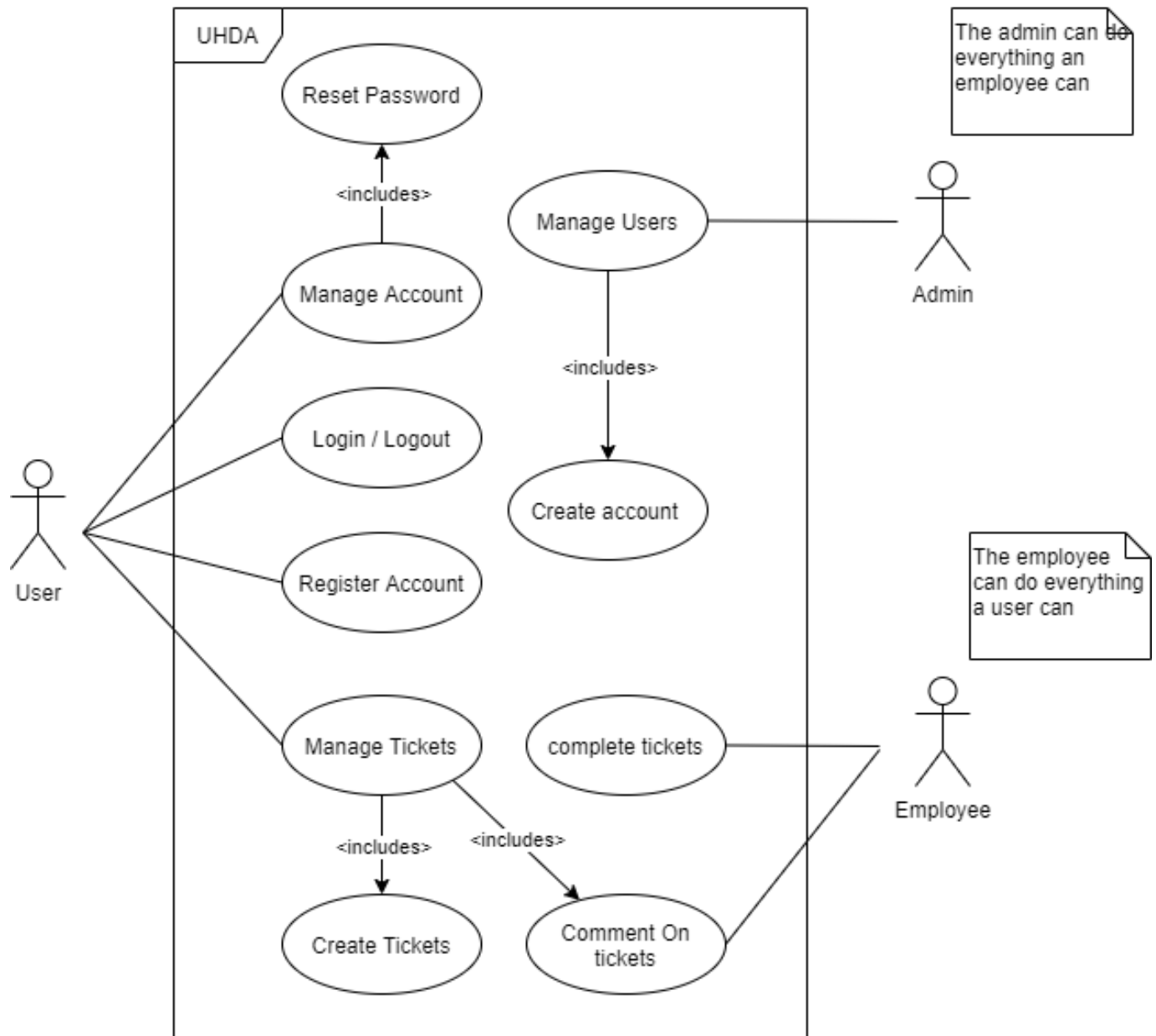
Systems that are similar on the market include *Hubspot*, *Jira Service Desk*, or *Zendesk*. All of the solutions above have a similar primary focus on IT but have complexed integrations or are merely bundled as part of a suite of enterprise applications. **UHDA** is a standalone solution that will focus on the bread-and-butter of IT to enhance the user experience.

The list below outlines a few of the major features of **UHDA**:

- **Constant Access, Daily:** 24/7 ability to create helpdesk tickets to your IT department once logged in through the webapp
- **Departmental Control:** As a department administrator, own your IT department's helpdesk and add the members allowed to create tickets
- **Ticket Categorization:** Professionals can sort available tickets by priority, resolution time, or issue type
- **Continuous Communication:** Users can comment or update tickets at any time, so that IT requests stay up to date and relevant
- **Slack Integration:** Connecting a workspace on slack will enable users to directly communicate with the professional assigned to their ticket via the Slack app

## Use Cases

### UHDA Use Case Diagram



## Group Dynamics

Group roles were decided as follows:

- Team Lead/Developer: Gregory Pohlod
- Release Manager/Acceptance Tester: Jacob Pauls
- Core Developer/Wiki lead: Michael Wilson

Creation of our group dynamics process was quite smooth, with no real disagreements that arose during the process. There were more in-depth conversations on team lead with Jacobs coding/git hub experience leaning us towards appointing him towards that. However, after some conversation and based on feedback from Jacob, it was decided that Gregory would be the Team lead as he has more experience leading teams, managing projects, and keeping everyone in line for timelines. Other roles were assigned based on need and interest, and conversation that everyone is expected to ask for help when needed, provide help when asked, and assist in the review process. For communication we are in a discord channel together which we use for chat and voice/meetings as needed. We will also schedule two standups per week minimum for updates.

## Software Toolset

Primarily, the software stack for this project will exist within the *Java* and *SpringBoot* framework. The rest of the tools are listed below:

- **Frontend:** *JSP, Javascript, JQuery*
- **Database Technology:** Local *h2* databases using *JDBC* drivers
- **Version Control:** *git* and *GitHub*
- **Project Management:** *GitHub* 'Projects'/Kanban Board
- **Testing:** *JUnit*

Overall, these tools will help capture our project metrics because they integrate incredibly well with one another. In our team, we also all have different backgrounds and a varying level of experience in each of these technologies. As a result, we each have strengths to make up for other members weaknesses and our metrics will be more easily and more widely captured.

## Project Schedule

**Iteration 1 - (10/08/20 - 10/22/20)**

**Goal/Outcome - Basic administrative functionalities and user definition**

Feature	Team Member	Points
<b>Login/logout functionality</b> <ul style="list-style-type: none"><li>Enables users, workspace administrators, and IT professionals to login and begin use of the UHDA</li></ul>	<b>Michael Wilson</b>	<b>2</b>
<b>Password reset</b> <ul style="list-style-type: none"><li>Any user, regardless of their permission level, should be able to execute a password reset on their UHDA login</li></ul>	<b>Gregory Pohlod</b>	<b>4</b>
<b>Admin-Side User Management</b> <ul style="list-style-type: none"><li>Workspace administrators should have a space where they can add/remove members within a workspace and modify their workspace role/permissions</li></ul>	<b>Jacob Pauls</b>	<b>4</b>

**Iteration 2 - (10/22/20 - 11/12/20)**

**Goal/Outcome - Build out ticketing functionality on both the back and frontend**

Feature	Team Member	Points
<b>Basic Ticketing Functionality</b> <ul style="list-style-type: none"><li>Build out objects related to ticketing</li><li>Fields should include (but aren't limited to) things like description, priority, time opened, issue type, and issue status</li><li>Create form for users to create tickets which passes the created objects throughout the rest of the webapp</li></ul>	<b>Jacob Pauls</b>	<b>3</b>
<b>Ticketing Display</b> <ul style="list-style-type: none"><li>Create frontend GUI for both users and IT professionals</li></ul>	<b>Michael Wilson</b>	<b>4</b>

<ul style="list-style-type: none"> <li>• Users should have the ability to see only the tickets they've created</li> <li>• IT professionals should see the current helpdesk queue of active tickets</li> </ul>		
<b>Ticket Actions</b> <ul style="list-style-type: none"> <li>• Create the backend logic required to assign and resolve created tickets</li> <li>• Integrate logic to define where ticket information is stored once resolved, cancelled, or generally closed</li> </ul>	<b>Gregory Pohlod</b>	<b>3</b>

### ***Iteration 3 - (11/12/20 - 11/26/20)***

#### ***Goal/Outcome - Supplemental/Luxury Features***

<b>Feature</b>	<b>Team Member</b>	<b>Points</b>
<b>User Hardware Requests</b> <ul style="list-style-type: none"> <li>• Integrate the issue type: "Hardware Requests" with a backend database</li> </ul>	<b>Gregory Pohlod</b>	<b>3</b>
<b>Ticket Commenting/Interaction</b> <ul style="list-style-type: none"> <li>• Users and IT Professionals should be able to comment on, and interact asynchronously on the page for a given ticket in-app</li> </ul>	<b>Michael Wilson</b>	<b>3</b>
<b>Slack Integration</b> <ul style="list-style-type: none"> <li>• Enable Slack integration so that users and IT professionals alike can message each other within their workspace about a given ticket</li> </ul>	<b>Jacob Pauls</b>	<b>4</b>

### ***Additional Consideration for Iteration 3***

<b>Iteration Catch-Up/Contingency</b> <ul style="list-style-type: none"> <li>• Time allotted for any sort of catch-up, bugfixes, or revisiting required for features implemented in Iterations 1 and 2</li> </ul>	<b>Jacob Pauls</b> <b>Gregory Pohlod</b> <b>Michael Wilson</b>	<b>N/A</b>
---	--	------------

## **Appendix A - Meeting Minutes**

### **Meeting #1 – 10/05/20 – 180 Minutes:**

#### **Focused on git flow and preparing for the official project by completing Assignment #2**

- Members present: Gregory, Jake and Michael
- Discussed assignment 2, worked through build and using git to work on individual section and practiced push and pull. Did some error checking. Made first release and pushed to git hub.
- Assigned some deliverable to finish off assignment 2,
  - each work on own image model, and merge to development,
  - Jake to create release branch and push second release.
  - Greg will create template for wiki
  - Michael completed some bug testing
  - Everyone to answer their own 3 questions on wiki
  - Group to meet on the 6<sup>th</sup> to review, create kanban board together and work through project proposal.

### **Meeting #2 – 10/06/20 – 90 Minutes:**

#### **Finished, and performed mini-retrospective on Assignment #2. Assigned project proposal tasks along with working together through several items.**

- Members present: Gregory, Jake and Michael
- Went through final bug tested version of assignment 2, everything works. Jake will merge and push release tomorrow after generated javadocs
- Created cover page and started project scope.
- Building out of Kanban board
- Created milestones and issues, and labels
- Transferred notes to issues and assigned label and assignees
- Group roles discussed
  - Greg Team Lead/Developer
  - Jake Release Manager/Acceptance Tester
  - Michael Core Developer/Wiki lead
- Discussed project scope, project in general and discussed application items and which iteration they would be focused on so we can work on use cases and project schedule
- Added assignees to kanban board and decided who will be primary on each issue

- Assistance from all team members when needed and review of final proposal is an expectation.
- Ended meeting at 8pm, all members available through discord throughout tomorrow for anything necessary.