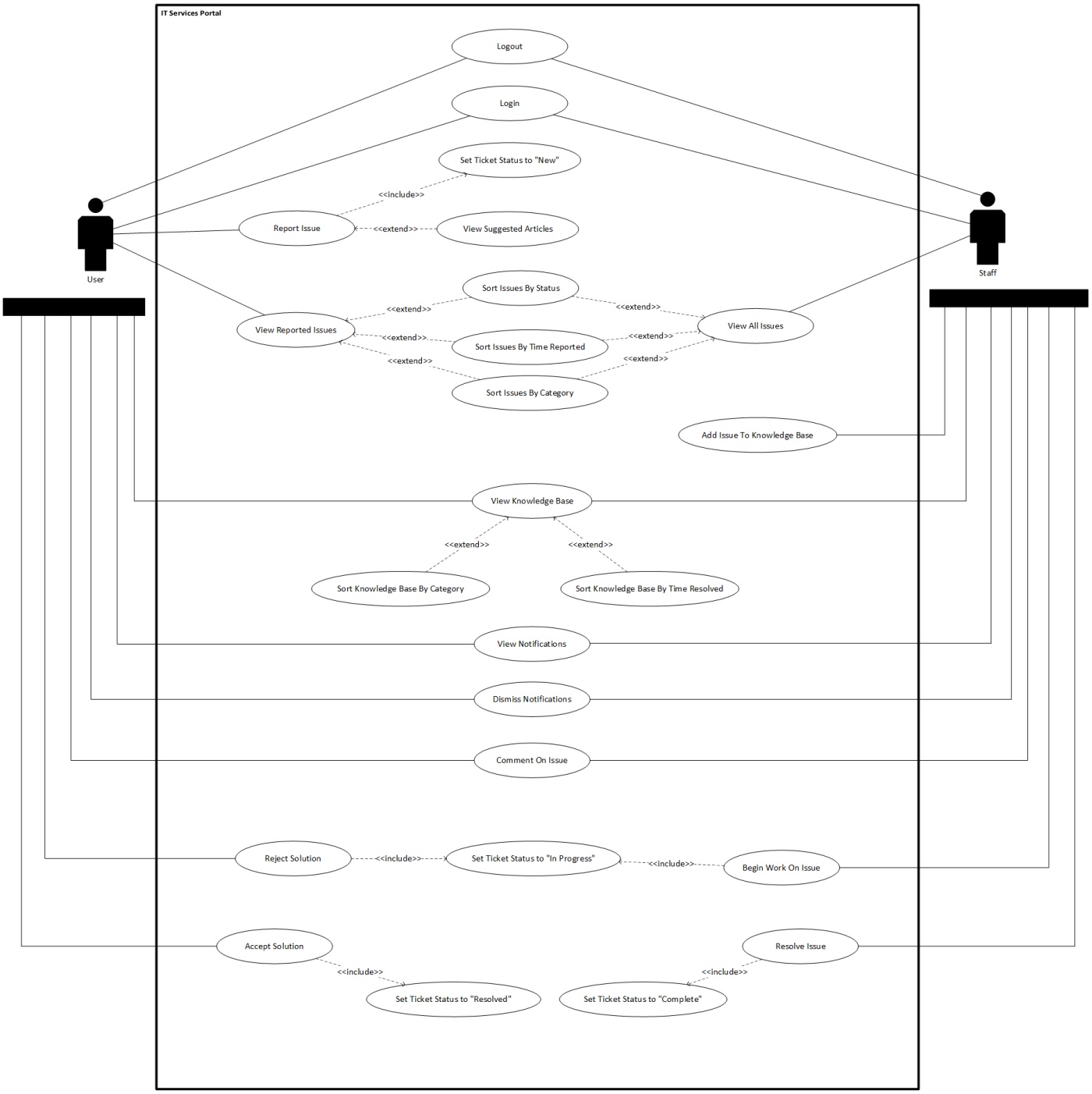
SENG2050 Web Engineering - Assignment 3

C3180044_C3281849_C3237808_FinalProject

Draft Documentation

CONTENTS

Use Case Diagram



USE CASE DESCRIPTIONS

Use Case Name:	Login
Description:	User/Staff login into the system to access features relevant to the IT Support System. Once the User or Staff has logged in with their email and password, they will be presented with their account role's service portal.
Actor:	User, Staff
Preconditions:	1. Must have valid account
Postconditions:	1. System displays service portal
Main Flow:	1. The user enters their email and password 2. The user submits email and password 3. The system validates the email and password 4. The system displays the service portal
Alternative Flow:	3a Incorrect email and password 1. Prompts user for email and password 2. Use Case resumes at main flow step 1

Use Case Name:	Logout
Description:	User/Staff logout of the system once they are done. Once logged out, the user will be presented with the login page if they wish to login again.
Actor:	User, Staff
Preconditions:	1. User must have valid account 2. User must be logged into account
Postconditions:	1. Display Login Page
Main Flow:	1. User clicks on Logout button 2. The system displays Login page

Use Case Name:	Report Issue
Description:	Users report an issue they are having. Once clicked on, the user is presented with a form that will request all information regarding the issue they are experiencing. The form is then sent to the IT Staff for Inspection.
Actor:	User
Include Use Cases:	1. Set ticket status to "New"
Preconditions:	1. The User is logged in
Postconditions:	1. Update Support Tickets 2. Notify IT Staff 3. Return to Portal
Main Flow:	1. User clicks report issue 2. User fills in form 3. (Extension Point: View suggested articles) 4. User submits form 5. Form is validated 6. Issue is sent to IT Staff 7. Issue Status is set to "New" 8. The system displays the portal
Alternative Flows:	4a Invalid or Empty form 1. Prompt user to fill form 2. Repeat from main flow 2

USE CASE DESCRIPTIONS

Use Case Name:	Set Ticket Status to “New”
Description:	Sets the status of a newly created issue to “New” and awaits for IT Staff to Begin work on issue.
Actor:	User
Preconditions:	1. The user has submitted an Issue
Postconditions:	2. IT Staff is notified
Main Flow:	1. Status of Issue changed to “New”

Use Case Name:	View Suggested Articles
Description:	User sees Knowledge Base articles that are related to the issue they are submitting by searching for keywords.
Actor:	User
Extend Use Cases:	1. Report Issue
Preconditions:	1. The user types in the title
Postconditions:	1. Suggest Articles from Knowledge Base
Main Flow:	1. The system takes in the title keywords 2. The system retrieves Knowledge Base articles from DB that contain keywords 3. The system displays the Articles

Use Case Name:	View Reported Issues
Description:	User views all issues that they have created. Once the user clicks on the view reported issue button, they will be redirected to the Issue page. The user is then allowed to sort by Time reported, Category or Status.
Actor:	User
Preconditions:	1. The user has logged in 2. The user has submitted issues
Postconditions:	1. Show Issue
Main Flow:	1. The user clicks on the View Reported Issue Button 2. The system retrieves user Issues from Database 3. The system displays the user Issues 4. (Extension Point: Sort Issues by Status) 5. (Extension Point: Sort Issues by Time Reported) 6. (Extension Point: Sort Issues by Category)
Alternative Flow:	2a No Reported Issue 1. Alerts user of no reported Issue 2. Use case resumes at main flow step 1.

USE CASE DESCRIPTIONS

Use Case Name:	View All Issues
Description:	Allows the IT Staff to view all the issues in the System. Once they click on the view issues button, they are redirected to the Issue page where the issues are displayed. They are also allowed to sort the issues in the manner that they like. This includes sorting by Time Reported, Category or by Status.
Actor:	IT Staff
Preconditions:	1. The staff has logged in 2. 2. Issues have been reported
Postconditions:	1. Show reported Issues
Main Flow:	1. The IT Staff clicks on View all Issues button 2. The system retrieves the user Issues from Database 3. The system displays the User Issues 4. (Extension Point: Sort Issues by Status) 5. (Extension Point: Sort Issues by Time Reported) 6. (Extension Point: Sort Issues by Category
Alternative Flows:	2a No Issues 1. Alert Staff of No Issues 2. Use case resumes at main flow step 1.

Use Case Name:	Sort Issues by Status
Description:	Allows both the User and Staff to sort the issues they want to view by Status.
Actor:	User, IT Staff
Extend Use Cases:	1. View Reported Issues 2. View All Issues
Preconditions:	1. Issues have been reported
Postconditions:	1. Sorted Issues by Status
Main Flow:	1. The system receives the request to sort by Status 2. The system request issues from Database 3. The system sorts issues by Status 4. The system displays sorted Issues by Status

Use Case Name:	Sort Issues by Time Reported
Description:	Allows both the User and Staff to sort the issues they want to view by the time reported.
Actor:	User, IT Staff
Extend Use Cases:	1. View Reported Issues 2. 2. View All Issues
Preconditions:	1. Issues have been reported
Postconditions:	1. Sorted Issues by Time Reported
Main Flow:	1. The system receives request to sort by time reported 2. The system requests issues from database 3. The system sorts issues by time reported 4. The system displays sorted issues by time reported

USE CASE DESCRIPTIONS

Use Case Name:	Sort Issues by Category
Description:	Allows both the User and Staff to sort the issues they want to view by Category.
Actor:	User, IT Staff
Extend Use Cases:	1. View Reported Issues 2. View All Issues
Preconditions:	1. Issues have been reported
Postconditions:	1. Sorted Issues by Category
Main Flow:	1. The system receives the request to sort by Category 2. The system request issues from Database 3. The system sorts issues by Category 4. The system displays sorted Issues by Category

Use Case Name:	Add Issue to Knowledge Base
Description:	Allows the IT Staff to add an issue that is set to “Resolved” or “Completed” to the Knowledge Base. By opening the issue and clicking the Add to KB Button, the issue will now be available in the Knowledge Base for others to view.
Actor:	IT Staff
Preconditions:	1. Issue is “Completed” or “Resolved”
Postconditions:	1. Return to Issues
Main Flow:	1. IT Staff clicks button to add Issue to KB 2. System generates copy of Issue 3. System adds copy to Knowledge Base 4. System displays Issue Page

Use Case Name:	View Knowledge Base
Description:	Users view the collection of Articles that are available in the Knowledge Base which may assist in any problems the user is having. The knowledge base is also able to be sorted to assist in which category the user is having trouble in.
Actor:	User, IT Staff
Preconditions:	1. The user is logged in 2. The Knowledge Base is populated with Articles
Postconditions:	1. Show Knowledge Base articles
Main Flow:	1. The user clicks on the View Knowledge Base button 2. The system requests All KB articles from database 3. (Extension Point: Sort Knowledge Base by Category) 4. (Extension Point: Sort Knowledge Base by Time Resolved) 5. The system displays all Knowledge Base Articles
Alternative Flows:	2a No Knowledge Base Articles 1. Alert user of empty KB 2. Use case resumes at main flow step 1

USE CASE DESCRIPTIONS

Use Case Name:	Sort Knowledge Base by Category
Description:	Allows both the User and Staff to sort the Knowledge Base Issues by their respective category.
Actor:	User, IT Staff
Extend Use Cases:	1. View Knowledge Base
Preconditions:	1. The Knowledge Base is not empty
Postconditions:	1. Sorted Knowledge base by Category
Main Flow:	1. The system receives request to sort by Category 2. The system requests KB issues from database 3. The system sorts issues by category 4. The system displays sorted issues by category

Use Case Name:	Sort Knowledge Base by Time Resolved
Description:	Allows both the User and Staff to sort the Knowledge Base Issues by the time resolved.
Actor:	User, IT Staff
Extend Use Cases:	1. View Knowledge Base
Preconditions:	1. The Knowledge Base is not empty
Postconditions:	1. Sorted Knowledge base by Time Resolved
Main Flow:	1. The system receives request to sort by Time Resolved 2. The system requests KB issues from database 3. The system sorts issues by time resolved 4. The system displays sorted issues by time resolved

Use Case Name:	View Notifications
Description:	Allows both the User and Staff to view notifications that have been assigned to them. By clicking on the notification, the user is taken to the Ticket where they can then view or edit.
Actor:	User, IT Staff
Preconditions:	1. User has been assigned a notification
Postconditions:	1. The ticket is displayed
Main Flow:	1. The User clicks on the notification tab 2. The system retrieves notifications from the Database 3. The system displays the Notifications 4. The user clicks on a notification 5. The system retrieves the ticket from the DB 6. The system displays the ticket

Use Case Name:	Dismiss Notifications
Description:	Allows both the User and Staff to dismiss notifications that have been assigned to them. By clicking on the notification, the user is shown the notifications and is then allowed to dismiss them.
Actor:	User, IT Staff
Preconditions:	1. User has been assigned a notification
Postconditions:	1. None
Main Flow:	1. The user clicks on the notification tab 2. The system retrieves notifications from the Database 3. The system displays the notifications 4. The user clicks on X, dismissing the notification

USE CASE DESCRIPTIONS

Use Case Name:	Comment on Issue
Description:	Allows the User or IT Staff to comment on an existing issue that has been opened. By clicking the button on the Issue page, the User is able to leave a comment on the Issue.
Actor:	User, IT Staff
Preconditions:	1. An Issue has been opened
Postconditions:	1. If Staff comment, Notify User
Main Flow:	5. User views Open Issue 6. User Clicks on Comment Button 7. User is presented with comment form 8. User enters information into form 9. User submits form 10. System validates form inputs 11. System adds form input to Issue 12. If User is Staff, system notifies User
Alternative Flows:	6a Incorrect Form Info 1. Prompt user to re-enter information 2. Use case resumes at main flow step 3

Use Case Name:	Reject Solution
Description:	Allows the user to reject a solution that has been marked as "Completed" by IT Staff allowing the issue to go back to "In Progress"
Actor:	User
Include Use case:	1. Set Ticket Status to "In Progress"
Preconditions:	1. Issue has been marked as "Completed"
Postconditions:	1. Issue marked as "In Progress"
Main Flow:	1. The user clicks Reject Solution 2. Issue Status is set to "In Progress"

Use Case Name:	Begin Work on Issue
Description:	Allows the staff to change the Status of an Issue from "New" to "In Progress" and allow comments from both the user and staff
Actor:	IT Staff
Include Use Cases:	1. Set Ticket Status to "In Progress"
Preconditions:	1. Issue is marked as "New"
Postconditions:	1. Issue is marked as "In Progress" 2. User is notified
Main Flow:	1. IT Staff clicks Begin Issue Button 2. Issue Status is set to "In Progress" 3. The system notifies User

Use Case Name:	Set Ticket Status to "In Progress"
Description:	Sets the Ticket status to "In Progress" and allows comments from both the User and IT Staff
Actor:	User, IT Staff
Preconditions:	1. The user has submitted an Issue
Postconditions:	1. Issue is "In Progress"
Main Flow:	1. Status of Issue changed to "In Progress"

USE CASE DESCRIPTIONS

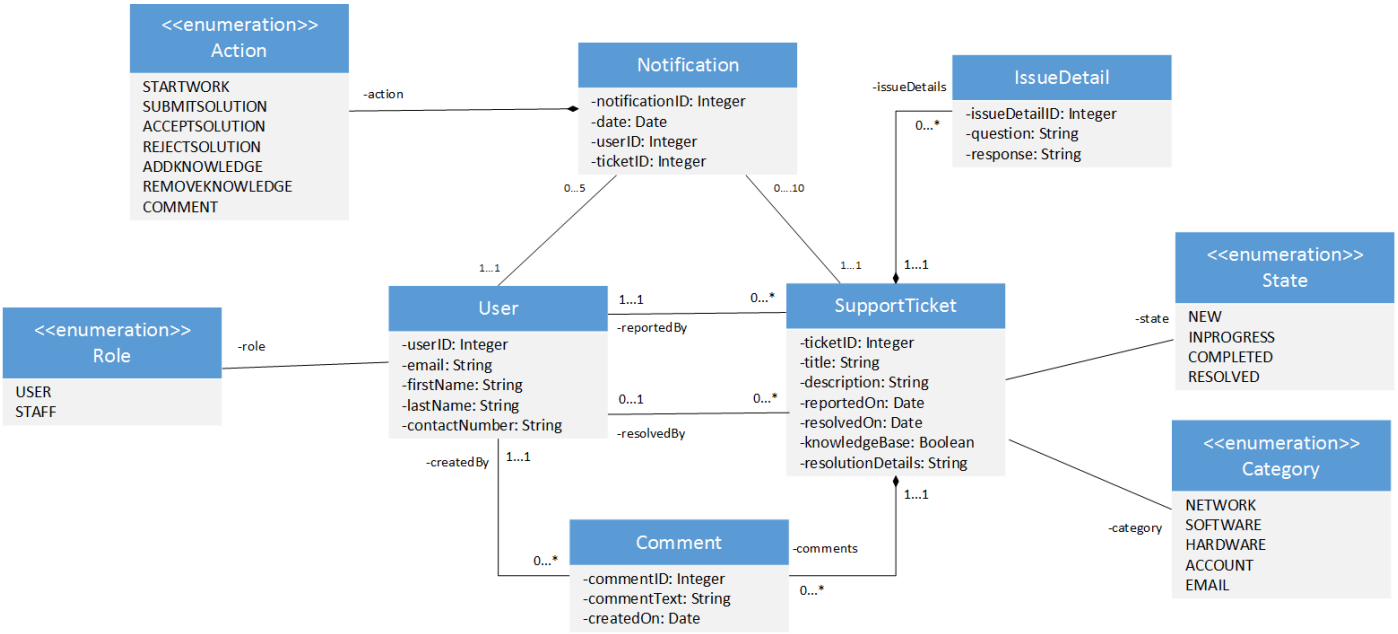
Use Case Name:	Accept Solution
Description:	Allows the user to accept a solution for the issue, marking the issue as “Resolved” and closing the issue.
Actor:	User
Include Use Cases:	1. Set Ticket Status to “Resolved”
Preconditions:	1. Issue has been marked at “Completed”
Postconditions:	1. Issue marked as “Resolved” 2. Staff is notified
Main Flow:	1. The user clicks Accept Solution 2. Issue Status is set to “Resolved” 3. The system notifies Staff

Use Case Name:	Set Ticket Status to “Resolved”
Description:	Sets the Ticket status to “Resolved” and closes the Issue so that no party can comment on it.
Actor:	User
Preconditions:	1. The issue is marked as “Completed”
Postconditions:	1. Issue marked as “Resolved”
Main Flow:	1. Status of issue changed to “Resolved”

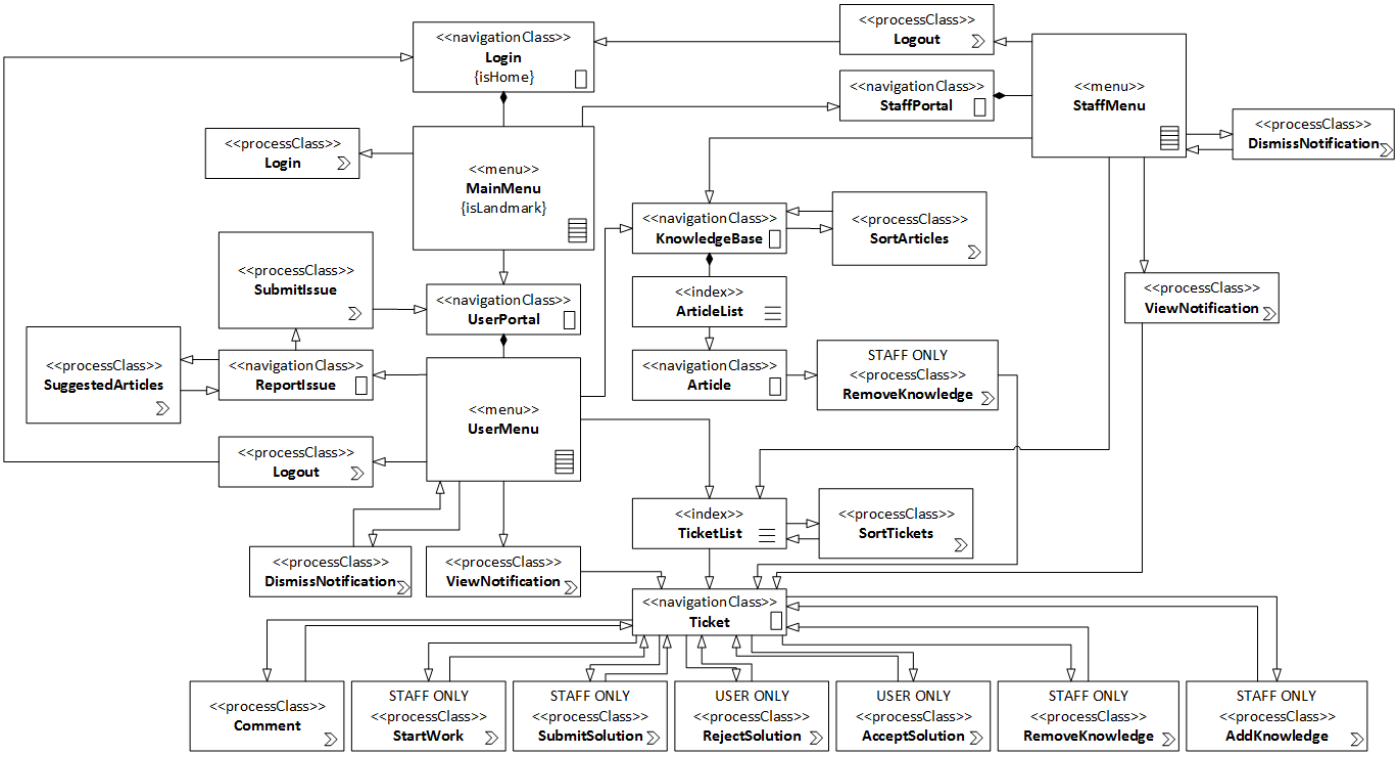
Use Case Name:	Resolve Issue
Description:	Allows IT Staff to resolve the issue, marking it as “Completed” and waiting for the User to either Accept or Reject the solution.
Actor:	IT Staff
Include Use Cases:	1. Set Ticket Status to “Complete”
Preconditions:	1. Issue has been marked as “In Progress”
Postconditions:	1. Issue marked as “Complete” 2. User is notified
Main Flow:	1. The Staff clicks Complete Issue 2. Issue status set to “Complete” 3. The system notifies User

Use Case Name:	Set Ticket Status to “Complete”
Description:	Allows the IT Staff to change the status of an Issue to “Complete”
Actor:	IT Staff
Preconditions:	1. The issue has been marked as “In Progress”
Postconditions:	1. Issue marked as “Complete”
Main Flow:	1. Set Issue status to “Complete”

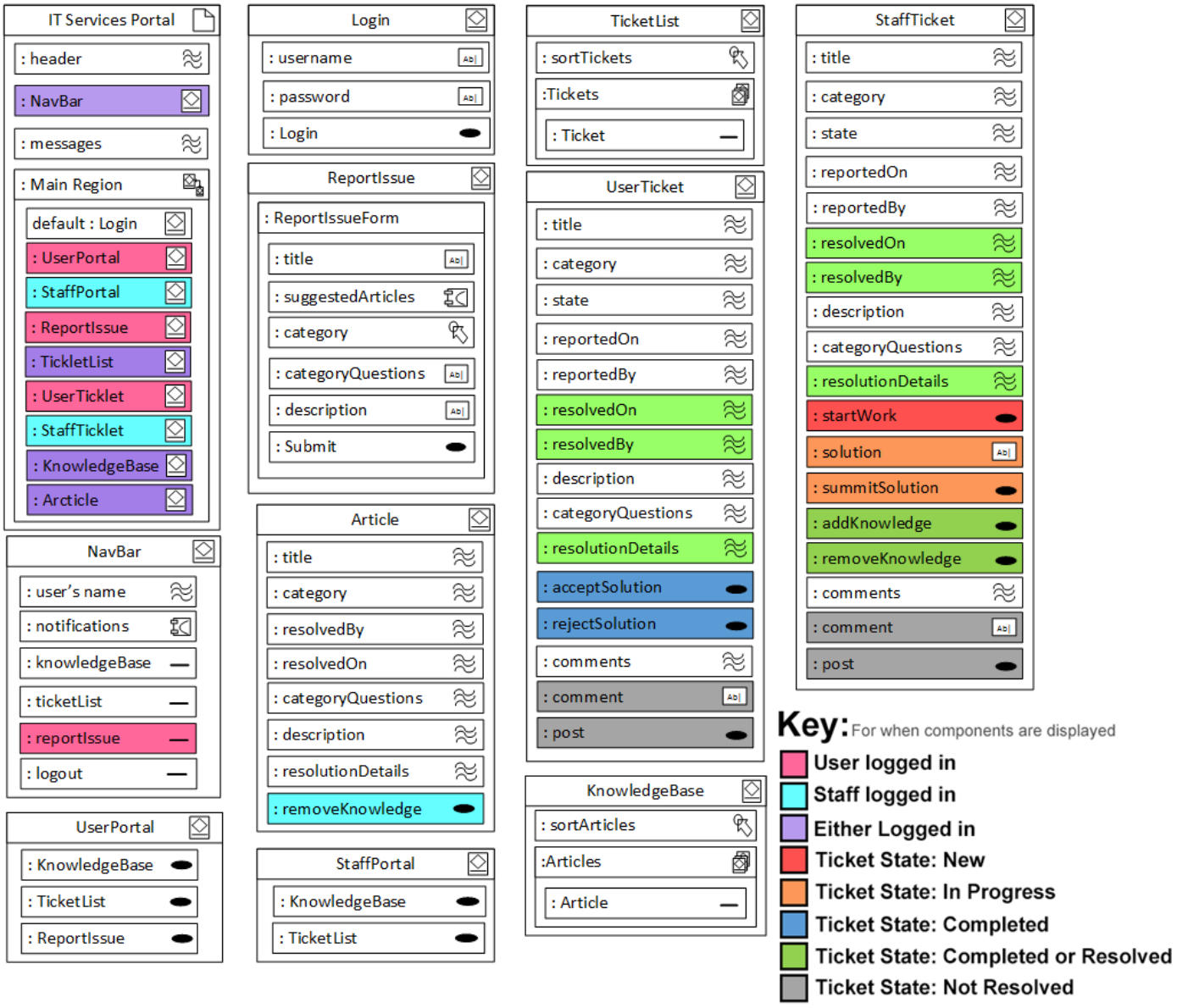
CONTENT MODEL



NAVIGATION MODEL



PRESENTATION MODEL



DEVELOPMENT PLAN

Task Allocation

We have discussed our group strengths and assigned tasks accordingly.

- Brice Purton will primarily work on the View
- Jonathan Williams will primarily work on the Database and Data Access Model
- Wajdi Younes will primarily work on the Controller and Model

However, tasks may require multiple input and assistance from other members. As this is a group assignment we'll be sharing a lot of the workload and tasks to make sure our web application integrates seamlessly.

Integration Plan

We will be using Git version control with a private GitHub repository from easily collaboration on our project. Development will be done in an MVC design pattern allowing each member to work separately on different concerns without preventing progress in other areas. Tasks will be broken down into logical groupings and worked on sequentially by each group member where the concern matches their primary task allocation. Once each task group is completed it will be integrated into the application then work on the next task group will commence.

Time Line

Task Name	Duration	Start	Finish	Resource Names
Develop Main Page Views	2 days	Mon 14/05/18	Tue 15/05/18	Brice
Develop Beans	4 days	Mon 14/05/18	Thu 17/05/18	Wajdi
Create & Populate Database	1 day	Mon 14/05/18	Mon 14/05/18	Jonathan
Develop User Data Access Model	1 day	Tue 15/05/18	Tue 15/05/18	Jonathan
Develop Authentication/Security Filter	1 day	Wed 16/05/18	Wed 16/05/18	Brice
Develop Log in/ Log Out Controller	1 day	Wed 16/05/18	Wed 16/05/18	Jonathan
Develop Log in/ Log Out View	1 day	Wed 16/05/18	Wed 16/05/18	Brice
Develop Ticket Data Access Model	8 days	Thu 17/05/18	Thu 24/05/18	Jonathan
Develop Issue Details Data Access Model	5 days	Thu 17/05/18	Mon 21/05/18	Jonathan
Develop Report Issue Controller	5 days	Thu 17/05/18	Mon 21/05/18	Wajdi
Develop Report Issue View	1 day	Thu 17/05/18	Thu 17/05/18	Brice
Develop Ticket List View	1 day	Thu 17/05/18	Thu 17/05/18	Brice
Develop TicketList Controller	2 days	Fri 18/05/18	Sat 19/05/18	Brice
Develop Comment Data Access Model	2 days	Sun 20/05/18	Mon 21/05/18	Jonathan
Develop View Ticket View	1 day	Sun 20/05/18	Sun 20/05/18	Brice
Develop Ticket Controller	2 days	Sun 20/05/18	Mon 21/05/18	Wajdi
Develop Knowledge Base View	1 day	Mon 21/05/18	Mon 21/05/18	Brice
Develop Knowledge Base Controller	3 days	Mon 21/05/18	Wed 23/05/18	Wajdi
Develop Article View	1 day	Wed 23/05/18	Wed 23/05/18	Brice
Develop Article Controller	1 day	Wed 23/05/18	Wed 23/05/18	Wajdi
Develop Notification System	3 days	Thu 24/05/18	Sat 26/05/18	Brice
Revise Database table data types and lengths	2 days	Tue 22/05/18	Wed 23/05/18	Jonathan
Insert final database testing data	2 days	Thu 24/05/18	Fri 25/05/18	Jonathan
Conduct Testing	14.88 days	Sun 27/05/18	Fri 8/06/18	All
Fix any bugs found	14.88 days	Sun 27/05/18	Fri 8/06/18	All
Due Date	1 day	Fri 8/06/18	Fri 8/06/18