**i-5 Computer Project**

**Project Title**: Job Ticket Manager

**Requirements**:

Phase 1:

1. Application needs to communicate with a data base and store tickets in said database for later retrieval.
2. When using the application, users are able to enter information into a textbox to which the information will be saved onto a job ticket
3. Job tickets can be printed out after they are loaded or created.
4. Job tickets can be saved, loaded, created, modified, and deleted
5. Job ticket format can be created. For example, if the owner wants to create a job ticket with a checkbox and 2 textboxes that can receive information, they should be able to do so.
6. Tickets are classified into 3 categories and sorted as such, not started, in progress, and complete.
7. Tickets are able to be marked as complete, not started, or in progress.
8. One database is required for this phase and the application keeps track of the database it’s attached to.

Phase 2:

1. A login system that keeps track of users and their tickets. Possibly a system that searches up users and filters by user instead.
2. The application supports looking up job tickets by name
3. The application supports looking up job tickets by status, in progress, not started, or complete
4. Multiple databases can be accessed with the application and switching databases loads in content from that database.

Phase 3:

1. Move this system from a local system onto the i-5 website for users to remotely send job tickets.
2. Job tickets entered from the website go to the designated database.
3. Email company email when a job ticket is sent from the website to the data base

**Project Model**: Evolutionary Model

Each phase can be split into its own prototype and completed as such.

**Phase 1 Brainstorming:**

-Splitting this into classes, there are parts that we need to keep track of.

1) **Database** (Singular)

2) **Active Ticket Template**

3) **Job Tickets** (Class)

- Document layout

# Information holder

+ Textboxes

~ Textbox entries

~ Textbox position

~ Textbox name

~ Textbox width, height

~ Textbox font size

+ Checkboxes

~ Checkbox position

~ Checkbox name

~ Checkbox size

~ Status (Checked or unchecked)

+ Required (Whether the information is required to contain an entry or not)

* Status (In-progress, not started, and complete)
* Customer name
* Ticket name
* Document template (The document that the ticket is based on. Set to a blank document by default)

4) **User Interfaces**

- **Job tickets**

**+** Change ticket status

+ Display all job tickets sorted by status

~ Ticket customer name

~ Ticket name

~ Ticket document

~ Ability to scroll

+ Change database

+ Print job ticket

- **Template window**

+ Load template (Change active template)

+ Create template

+ Edit template

~ Add, modify, or remove textbox

# create textbox

# modify position

# modify size

# delete textbox

~ Add, modify, or remove checkbox

# create checkbox

# modify position

# delete checkbox

+ Save template

* **Create ticket**

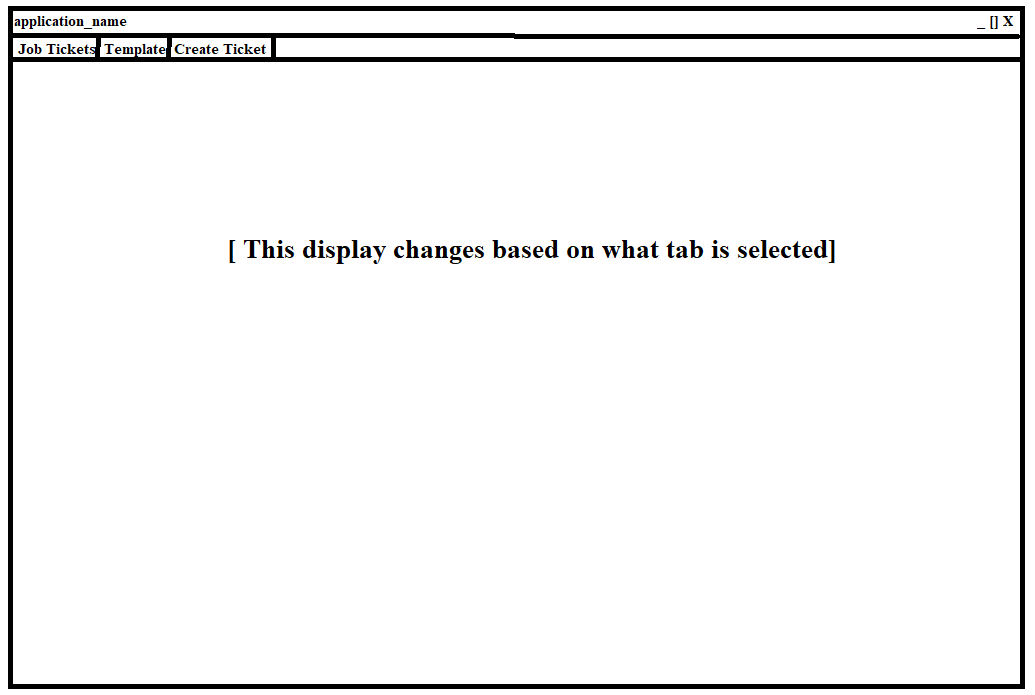
+ Display entry boxes

+ Allow customer to enter entries

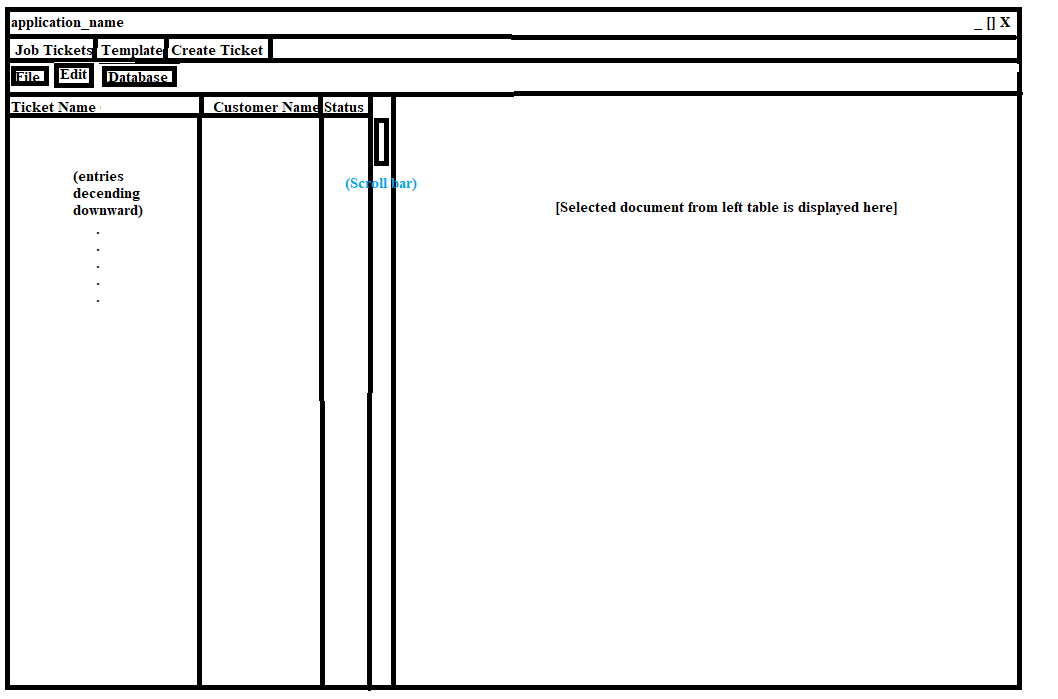
+ Save ticket to current database. Saved as not started.

**UI Sketches**

**Main Screen**

****

**Job Tickets Screen**

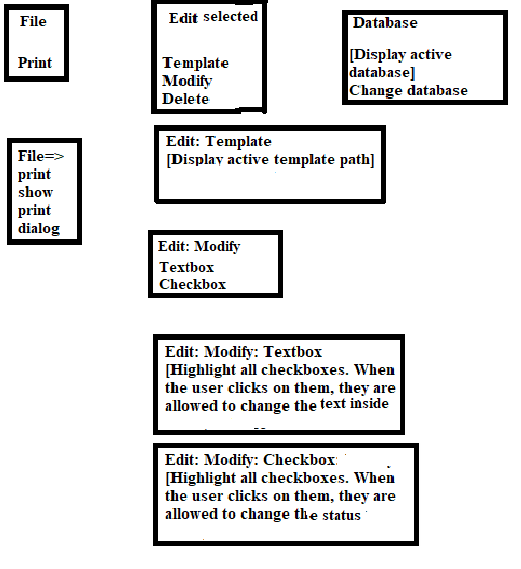
****

**- Status is allowed to be changed through a combination box**

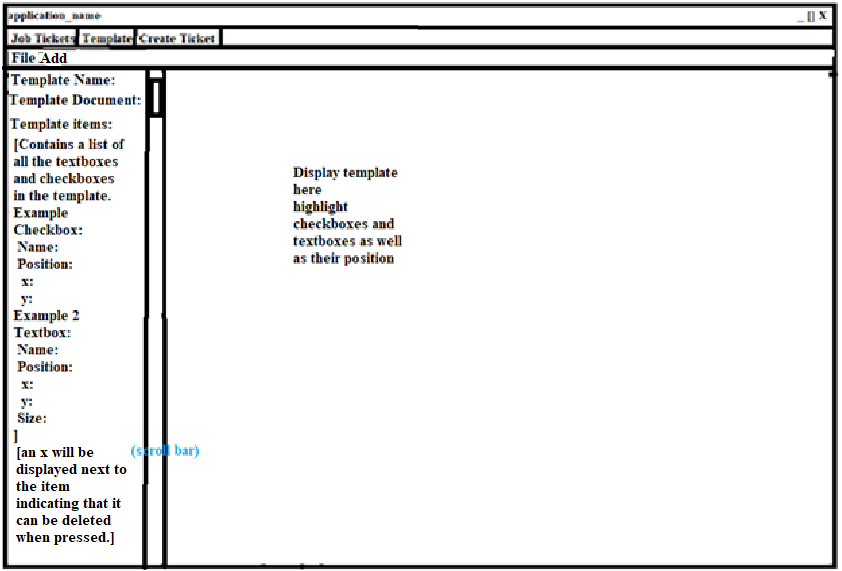
**- Name is allowed to be changed through a textbox**

**- Customer is allowed to change through a textbox**

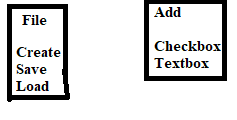
**-Job ticket drop down menus**

****

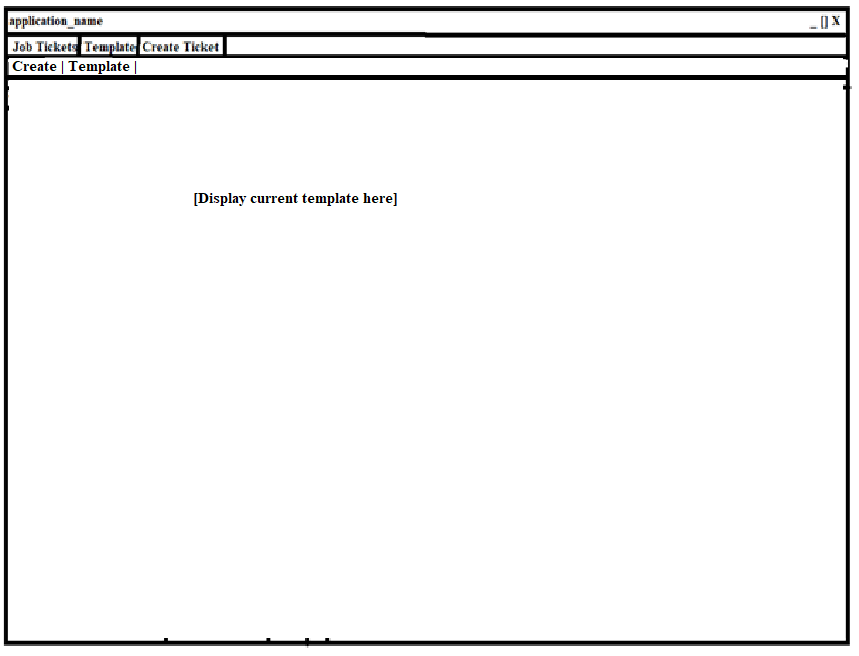
**Template Screen**

****

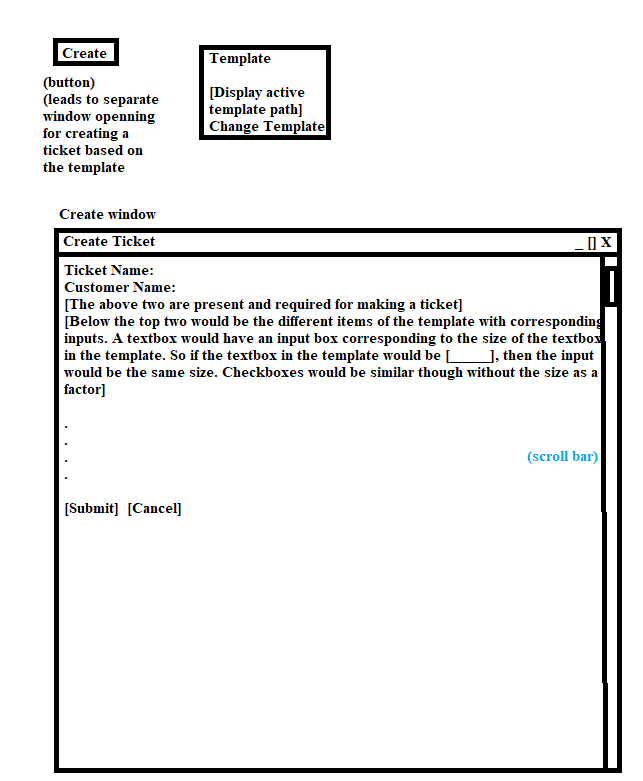
**-Template menu items**

****

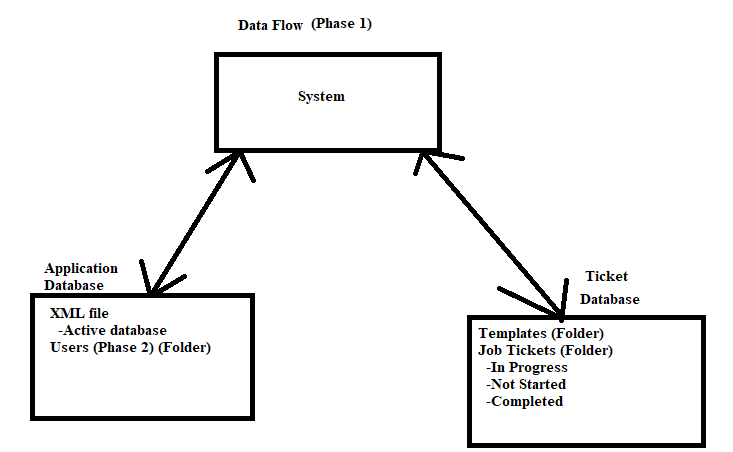
**Create Ticket Screen**

****

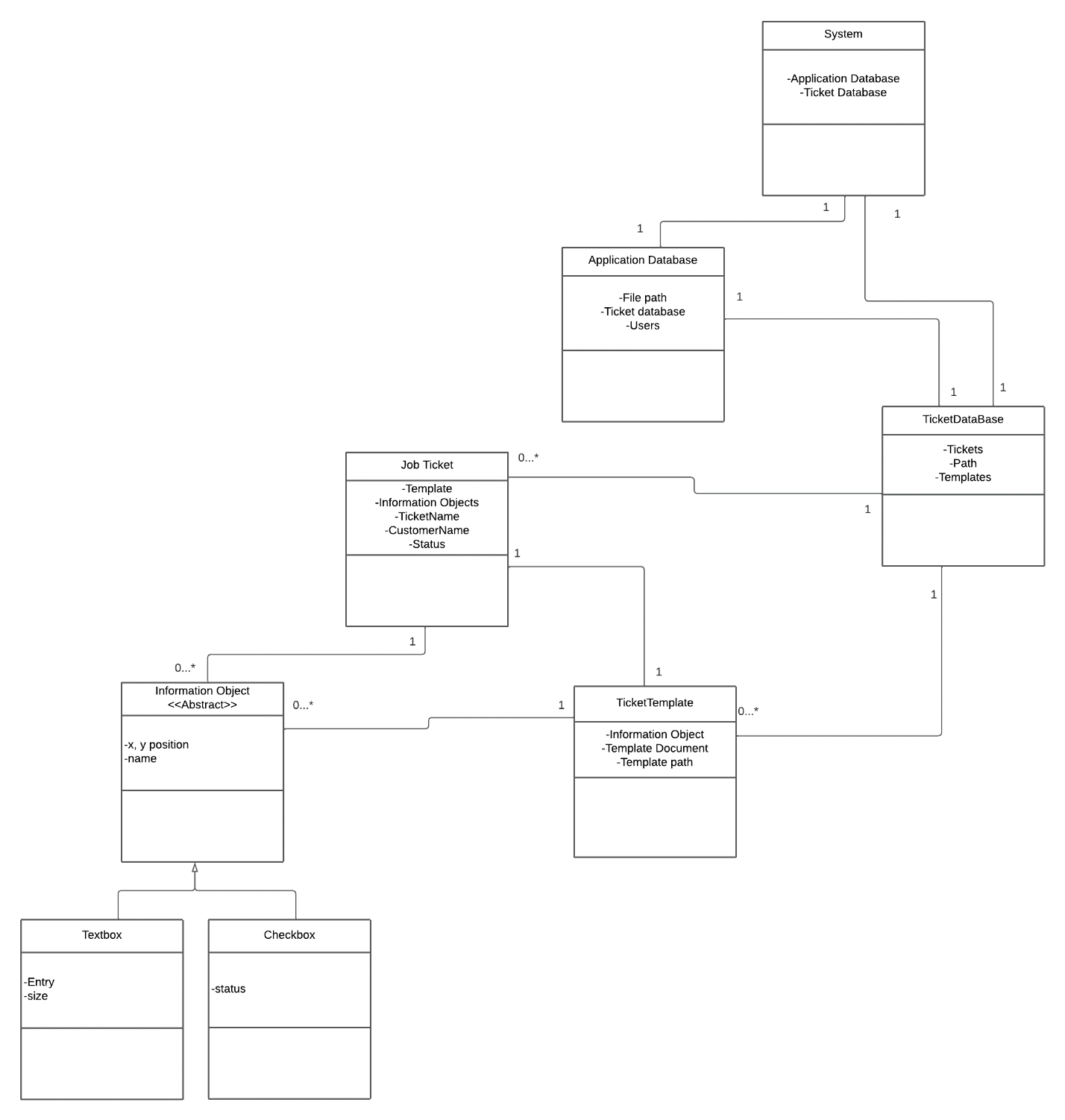
**-Create screen menu items**

****

**Data Flow (Phase 1)**

****

**Class Diagram (Phase 1)**

****

**Work Flow (Phase 1)**

1. Create code skeleton

-Set up first form with tabs

-Create engine project with relevant classes

+Job ticket

+Database

+Template

1. Implement functions for setting up and using a database
2. Implement functions for setting up a template

-Make, modify, delete template

-Set active template

-Default template

4) Implement functions for using templates to create a ticket

- Create ticket

- Store ticket in data base

5) Implement functions for editing job tickets in the data base

- Display job ticket

- Edit status

- Edit title

- Edit customer

- Edit textboxes and checkboxes

6) Implement functions for printing job tickets

7) Deploy phase 1